Apple DiSK
Il mensile con disco programmi per Apple II
Sped. in Abb. Postale Gr. III/70%
Anno III - Giugno/Luglio 88
N. 17/18 - L. 15.000



ATLETICA
Tutti campioni con AppleDisk

SENZA VELI
Il DOS 3.3 non ha più segreti

QUINDICI TRUCCHI
Le potenti routine per dar più vita ai programmi

BABY ENGLISH
Impararlo è un gioco col labirinto ortografico

NUMERI
Notazione scientifica a portata di mano

EXPLORER
Il programma che sa tutto del dischetto

ALIENI
I riflessi a dura prova

Gruppo Editoriale JCE

Il mensile con disco programmi per i personal computer MS-DOS

PC DISK Magazine

In questo numero:
1 DISCO PROGRAMMI
+
1 DISCO VERGINE
per back up

MAGGIO 1988 - Numero 20 - L. 15.000

Anno III - Sped. in Abb. Post. Gr. III/70%

Salute
Pronto soccorso sempre a posto

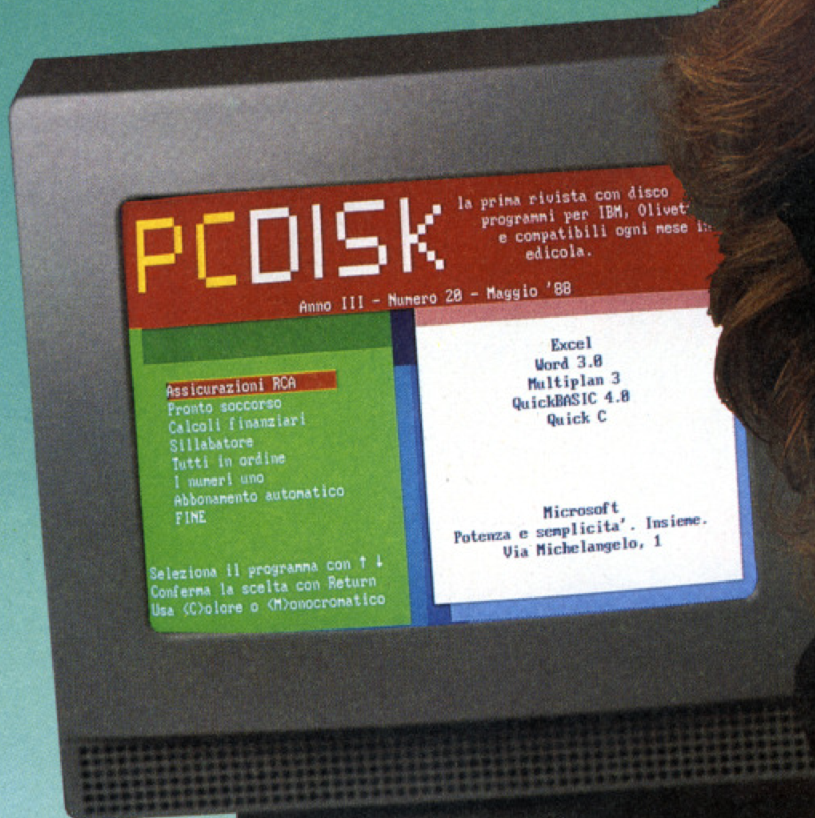
Affari & finanza
Conviene il mutuo casa?

Assicurazione
Tutti i conti dell'automobile

Sillabazione
Massima velocità

CD Rom
La memoria del futuro

Per chi comincia
Ecco le parole chiave



Prove software
Xerox Ventura 1.1
Microsoft Excel
Turbo Pascal 4.0

in tutte le edicole

Apple DiSK N°17/18

Direttore responsabile
Paolo Romani

Caporedattore
Francesca Marzotto

Responsabile grafico Desktop Publishing
Adelio Barcella

Impaginazione elettronica
Mirko Diani, Giorgio Meroni

Responsabile programmi
Giorgio Caironi

Collaboratori
Renato Gelforte, Mario Pettenghi, Dolma Poli

Segretaria di redazione
Patrizia Angelo

Gruppo Editoriale JCE srl
Sede legale, Direzione, Redazione, Amministrazione
Via Ferri 6 - 20092 Cinisello Balsamo (Mi)
Tel. 02/61.73.441-61.72.671
-61.72.641-61.80.228
Telex 352376 JCE MIL I - Telefax 61.27.620

Direzione Amministrativa
Walter Buzzavo

Pubblicità e Marketing:
Gruppo Editoriale JCE - Divisione Pubblicità
Via Ferri 6 - 20092 Cinisello Balsamo (Mi)
Tel. 02/61.20.586-61.27.827
61.23.397-61.29.0038

Abbonamenti e spedizioni
Daniela Radicchi

I versamenti vanno indirizzati a:
Gruppo Editoriale JCE, Via Ferri 6-20092 Cinisello B. (MI), mediante l'emissione di assegno circolare, cartolina vagliao utilizzando il c.c.p. n. 351205
Per i cambi di indirizzo allegare alla comunicazione l'importo di L. 1.000, anche in francobolli, e indicare insieme al nuovo anche il vecchio indirizzo.

APPLEDISK
Rivista mensile, una copia L. 15.000, numeri arretrati il prezzo di copertina.
Registrazione Tribunale di Milano N. 312 del 7.6.1986

Fotolito
Bassoli - Milano.

Stampa
GEMM Grafica srl, Paderno Dugnano (MI).

Diffusione
Concessionario esclusivo per l'Italia A.&G. Marco SpA, Via Fortezza 27 20126 Milano
Spedizione in abb. post. gruppo III/70.

Abbonamenti:
Annuale L. 150.000
(con in dono 20 dischetti vergini).

Sommario

Un consiglio pratico

Per una corretta gestione dei programmi di AppleDisk, è bene trasferirli su di un altro disco, o meglio ancora dividerli su più dischi, lasciando intatto quello originale.
In questo modo si otterranno inoltre due vantaggi: se un programma si dovesse danneggiare, se ne potrà rifare la copia, avendo conservato l'originale; se dovesse sorgere un motivo di reclamo presso il Gruppo Editoriale JCE, lo si potrà documentare inviando il disco originale intatto: quelli già manipolati dagli utenti, infatti, non potranno essere tenuti in considerazione.
Si tratta naturalmente di un'avvertenza puramente cautelativa, poiché ogni dischetto di AppleDisk è accuratamente controllato e, se utilizzato correttamente, non riserva alcuna sorpresa.

Novità

Full teflon jacket

Non c'è nulla da fare, anche il floppy dalle migliori caratteristiche magnetiche e strutturali, è in balia di utenti distratti che ne toccano la superficie con le dita, vi rovesciano liquidi di varia natura (preferibilmente caffè, durante i break).

Per proteggere i vostri dati da distrazioni di qualsiasi tipo, la Verbatim, società del gruppo Kodak, leader nel settore dei supporti magnetici per PC, ha presentato la linea Data-Life-Plus.

Cosa c'è di nuovo? Un'idea, che nella sua semplicità ha del geniale: l'intera superficie magnetica è stata ricoperta da uno strato di Teflon, in una particolare formula che ne garantisce resistenza e durata.

L'effetto è sorprendente: alla conferenza stampa, in cui sono stati presentati i nuovi floppy, abbiamo visto sporcare di marmellata (e poi ripulire) i poveri dischi, li abbiamo visti maltrattare con rossetti, bevande analcoliche e persino vernice, senza recare danni ai dati registrati.

Non è un successo casuale: prima della commercializzazione i dischi sono stati utilizzati in sede di test da oltre 1.200 utenti professionali selezionati dalla stessa Verbatim. Sulla base di questi risultati il progetto è stato perfezionato, fino al conseguimento del risultato attuale.

Molto buono anche il biglietto da visita tradizionale dei floppy: il materiale magnetico è di ottima qualità e inoltre viene offerta un'adeguata protezione antistatica battezzata DataHold II , che garantisce un'affidabilità ancora superiore.

Un superfloppy-disk, dunque? "Certo", ci conferma Daniel Bourns, presidente della Verbatim, "oggi i floppy-disk sono prodotti simili a tanti altri.

Per questo motivo, i nostri studi di marketing ci dicono che i tempi sono maturi per offrire ai consumatori un prodotto altamente qualificato che garantisca il massimo dell'operatività e dell'affidabilità, sotto tutti gli aspetti."



Per editori neofiti

Desktop publishing è per Mac ma se volete saperne di più questa notizia fa al caso vostro.

Se i vostri lavori eseguiti con Apple Edit non vi soddisfano, non potete certo biasimare nessuno. Parte delle vostre frustrazioni nasce sicuramente da carenze conoscitive dell'arte del comunicare impaginando. Aldus (la nota società americana che ha sviluppato Page Maker) ha pensato bene di aiutarvi pubblicando un prezioso manualetto: "La Guida Aldus al Basic Design" dove sono svelati molti segreti noti ai più bravi grafici impaginatori. Gli spazi bianchi, i riquadri, i caratteri vanno dosati e miscelati con gusto, e sono frutto di precise norme grafiche.

Viene suggerito come rendere ordinate le vostre pagine grazie a linee orizzontali e verticali o come usare con giusto equilibrio i negativi e gli sfondi.

Il volumetto consta di 68 pagine ed è stato ovviamente prodotto ed illustrato con Page Maker. L'autore Roger C. Parker, l'edizione italiana è stata curata da Otorino Baseggio, direttore responsabile di Italia Publisher, rivista dedicata al Desktop Publishing.

Il manualetto è stato reimpaginato sull'edizione originale da Iret System, e potete trovarlo presso i rivenditori Iret al prezzo di lire 6.000. Nel caso il rivenditore ne sia sprovvisto potete richiederlo direttamente alla Iret System, via Emilio Santo Stefano, 38 - 42100 Reggio Emilia.

È l'ora dell'Europa

Apple guarda all'Europa. Non solo per vendere più computer sui mercati del vecchio continente, ma anche per sfruttare meglio l'enorme serbatoio di intelligenza disponibile da questa parte dell'Atlantico. E per entrambi i motivi apre a Parigi il suo centro di ricerca e sviluppo. "È un elemento chiave nella strategia di rafforzamento della posizione della Apple in Europa," ha detto a Londra Mike Spindler presentando Edward Colby, che ne sarà il direttore. Colby ha lavorato a Cupertino, il quartier generale californiano della Apple, negli ultimi cinque anni, dove era responsabile delle strategie di prodotto con particolare riferimento alle Cpu, al software di sistema, alle periferiche e ai prodotti di comunicazione.

L'annuncio della fondazione del centro è avvenuto in occasione dell'evento "Information Systems and Technology" tenutosi a Londra il 21 marzo 1988 al Queen Elizabeth II Conference Center.

Il nuovo centro di Parigi lavorerà in modo indipendente ma in strettissimo collegamento col centro ricerche e sviluppo di Cupertino. Vi lavoreranno una quarantina di persone che Spindler conta di assumere fra i migliori ricercatori disponibili..



Un programma per archiviare i record dei vostri beniamini nelle gare di altetica, sci, nuoto, e tutte le altre specialità che desiderate.

Tutti i primati sotto controllo

Il programma Schedario gare di atletica leggera permette di archiviare i risultati ottenuti in determinati periodi dell'anno, in particolari località o manifestazioni e di effettuare ricerche su di essi.

Può essere facilmente adattato ad ogni altro tipo di sport. Prima di utilizzare il programma è consigliabile formattare un dischetto in DOS 3.3 per registrare il dati relativi all'archivio. Questo sarà il vostro disco dati.

Descrizione dell'archivio

L'archivio usato nel programma può essere considerato come un insieme di schede simili a quelle visibili in **figura 1.** In realtà è un file random (ad accesso casuale) formato di n record.

Ad ogni record del file Gare corrisponde una scheda dell'archivio.

Vediamo insieme quali sono i campi dei records del file Garein dettaglio:

- **Numero della scheda.** Corrisponde al numero del record.

- **Località** (20 caratteri).Questo campo può contenere al massimo 20 caratteri. Sarà sempre associato alla variabile SC$(1) (nel programma) e conterrà il nome della località dove si è svolta la gara alla quale la scheda si riferisce.

- **Data** (8 caratteri). Contiene 8 caratteri (4 per l'anno, 2 per il mese e 2 per il giorno). Quando nel programma si confrontano delle date, si considera tutta la variabile senza effettuare controlli particolari sull'anno, il mese e il giorno.

Per esempio, per verificare che 10 06 1984 > 22 04 1978, con il metodo sopra descritto si avrà: 1984,06,10 > 1978,04,22 (usando il Basic Applesoft) senza dover effettuare controlli su anno, mese e giorno.Il campo (nel programma) verrà associato alla variabile SC$(2).

- **Manifestazione** (39 caratteri). Può contenere al massimo 39 caratteri, verrà associato (nel programma) sempre alla variabile SC$(3) e contiene il nome della manifestazione nell'ambito della quale si è svolta la gara.

- **Cognome** (18 caratteri). Contiene al massimo 18 caratteri, verrà associato (nel programma) sempre alla variabile SC$(4) e contiene il cognome dell'atleta che ha partecipato alla gara alla quale la scheda (record) si riferisce.

- **Nome** (18 caratteri). Contiene al massimo 18 caratteri, verrà associato (nel programma) sempre alla variabile SC$(5) e contiene il nome dell'atleta che ha partecipato alla gara alla quale la scheda (record) si riferisce.

- **Specialità** (28 caratteri). Contiene al massimo 28 caratteri, verrà associato (nel programma) sempre alla variabile SC$(6) e contiene il nome della specialità in cui ha gareggiato l'atleta nella gara alla quale la scheda si riferisce.

- **Carattere di controllo** (1 carattere). Verrà associato (nel programma) sempre alla variabile SC$(7). Può contenere il valore 1, 2 o 0. Se è 0 significa che la scheda contiene una gara di marcia o di corsa e di conseguenza il risultato è espresso in ore, minuti primi, minuti secondi, centesimi di secondo. Se è 1 la scheda contiene una gara di salti o lanci (nel gergo dell'atletica leggera si chiamano concorsi) e il risultato è espresso in metri. Se è 2 la scheda contiene una gara di decathlon e di conseguenza il risultato è espresso in minuti.

- **Risultato** (8 caratteri). Può contenere al massimo 8 caratteri, verrà associato (nel programma) alla variabile SC$(8) e contiene il risultato della prestazione alla quale la scheda si riferisce.

- **Piazzamento** (5 caratteri). Può contenere al massimo 5 caratteri, verrà associato (nel programma) sempre alla variabile SC$(9) e contiene un numero che esprime l'ordine di classifica dell'atleta nella gara alla quale la scheda si riferisce.

Facendo la somma del massimo dei caratteri che possono essere contenuti nei campi di un record, si deduce che i records sono lunghi 145 caratteri.

Per memorizzare quanti records sono effettivamente presenti nel file Gare, viene creato dal programma un file sequenziale chiamato N Record e ogni volta che l'utente elimina o aggiunge una scheda all'archivio, viene aggiornata la variabile NR e memorizzata nel file N Record.

scheda n.

Località data
manifestazione
cognome nome
specialità
risultato piazzamento

Figura 1.

Tavola 1. Menù principale.

SCHEDARIO GARE DI ATLETICA LEGGERA

MENU':
[1] CREAZIONE SCHEDARIO
[2] INTRODUZIONE NUOVA SCHEDA
[3] ELIMINAZIONE
[4] CORREZIONE
[5] RICERCA
[6] STAMPA SU CARTA
[7] FINE
SCEGLI:

Procedure del programma

Il programma, dopo aver chiesto di inserire il floppy disk che deve contenere l'archivio nel drive 1, presenta all'utente il menù principale, visibile in **tavola 1**, che prevede 6 procedure più quella di Fine del programma. Esaminiamole ora una per una nei particolari.

1) Creazione schedario.

Come tutte le procedure, per prima cosa chiede all'utente la conferma per continuare (nel caso abbia sbagliato a scegliere procedura, può tornare al menù premendo M).

Questa procedura serve esclusivamente a creare il file sequenziale N Record e a memorizzare il valore 0 per informare il programma che ci sono 0 schede nell'archivio, dopodiché torna al menù principale.

Questa opzione va scelta solo quando utilizzate il programma per la prima volta.

2) Introduzione nuova scheda.

Dopo la conferma per continuare, il programma chiede di introdurre i campi del record (scheda) nel seguente ordine:

- Località dove si è svolta la gara da memorizzare. Non bisogna introdurre più di 20 caratteri.

- Data in cui si è svolta la gara. Bisogna scrivere prima il giorno (2 caratteri), poi il mese (2 caratteri) e infine l'anno (4 caratteri) tutti in cifre decimali e separati da virgole.

- Manifestazione nell'ambito della quale si è svolta la gara. Non bisogna introdurre più di 39 caratteri.

- Cognome dell'atleta che ha partecipato alla gara. Massimo 18 caratteri di lunghezza.

- Nome dell'atleta che ha partecipato alla gara. Massimo 18 caratteri.

- Specialità nella quale l'atleta ha gareggiato. Non deve essere più lunga di 28 caratteri.

Dopo l'input della specialità, viene chiesto all'utente di premere: 0 se la gara è di marcia o di corsa, 1 se la gara è di salti o lanci (concorsi), 2 se la gara è di decathlon.

- Risultato ottenuto dall'atleta nella gara alla quale la scheda si riferisce.

- Piazzamento (ordine di classifica) in cifre decimali.

Dopo aver introdotto tutti i dati da registrare, il programma chiede la conferma. Se si preme N il programma torna a chiedere l'introduzione dei campi del record, se si preme S chiede se si vuole introdurre un'altra scheda. Se si risponde anche a questa domanda premendo S, viene richiesta l'introduzione dei campi di un nuovo record (scheda), se si preme N torna al menù principale.

3) Eliminazione.

Dopo la conferma per continuare, il programma chiede all'utente il numero della scheda da eliminare (che va ricercata con la procedura 5 - Ricerca). Se si introduce un numero superiore al numero di schede presenti nell'archivio, il programma risponde con il messaggio "La scheda non esiste", altrimenti visualizza la scheda da eliminare e chiede se è quella.

Se si risponde N il programma torna a chiedere nuovamente il numero di scheda da eliminare, se si preme S la scheda viene eliminata e il programma torna al menù principale.

4) Correzione.

Dopo la conferma per continuare, il programma chiede all'utente il numero della scheda da correggere (che va eventualmente ricercata con la procedura 5 - Ricerca).

Se si introduce un numero maggiore del numero di schede contenute nell'archivio, il programma risponde con il messaggio "La scheda non esiste" e torna al menù principale dopo la pressione di un tasto qualsiasi, altrimenti visualizza la scheda da correggere e chiede la conferma.

Se si risponde N, torna a chiedere il numero della scheda da eliminare, se si risponde S chiede all'utente se vuole la stampa su carta della scheda (così da poterla controllare meglio prima di correggerla). Se si risponde anche a questa domanda premendo il tasto S, la scheda viene stampata su carta, dopodiché viene chiesto di reintrodurre la scheda da correggere. Dopo la conferma, il programma torna al menù principale.

5) Ricerca.

Dopo la conferma per continuare, viene presentato all'utente il menù Ricerca 1, visibile in **tavola 2**, che chiede su quale campo il programma deve effettuare la ricerca. Poi bisogna specificare cosa si ricerca.

Se si decide di effettuare ricerche su una specialità, il programma visualizza (a richiesta dell'utente stampa anche

Tavola 2. Menù Ricerca1.

SCHEDARIO GARE DI ATLETICA LEGGERA

SU COSA VUOI EFFETTUARE LA RICERCA?
[1] LOCALITÀ
[2] DATA
[3] MANIFESTAZIONE
[4] COGNOME
[5] NOME
[6] SPECIALITÀ
[7] PIAZZAMENTO

Tavola 3. Menù Ricerca 2.

SCHEDARIO GARE ATLETICA LEGGERA

COSA RICERCHI?
[1] MANIFESTAZIONE < GIOCHI OLIMPICI
[2] MANIFESTAZIONE <= GIOCHI OLIMPICI
[3] MANIFESTAZIONE = GIOCHI OLIMPICI
[4] MANIFESTAZIONE <> GIOCHI OLIMPICI
[5] MANIFESTAZIONE >= GIOCHI OLIMPICI
[6] MANIFESTAZIONE > GIOCHI OLIMPICI

su carta) le schede contenenti le gare delle specialità indicata, dopodiché torna al menù principale.

Se invece si decide di effettuare la ricerca su un qualsiasi altro campo, il programma propone all'utente un secondo menù, visibile in **tavola 3**, che chiede con quale criterio bisogna effettuare la ricerca. Dopo aver operato anche quest'ultima scelta, il programma visualizza (ed eventualmente stampa su carta) le schede che hanno le caratteristiche prescelte e a ricerca terminata torna al menù principale.

6) Stampa su carta.
Scegliendo questa opzione viene chiesto il numero della scheda da stampare. Se si introduce un numero maggiore del numero di schede contenute nell'archivio, il programma visualizza il messaggio "Questa scheda non esiste" e aspetta che venga premuto un tasto per tornare al menù principale, altrimenti mostra all'utente la scheda (su video) e chiede se è quella da stampare.

Se si risponde premendo N, il programma chiede nuovamente il numero della scheda da stampare, se si preme S la scheda viene stampata su carta e il programma visualizza il menù principale. La scheda da stampare può essere ricercata con l'opzione 5.

7) Fine.
Se si decide di terminare, il programma chiede un'ulteriore conferma. Se si risponde con N, il programma torna al menù pricipale, se si risponde premendo S il video di testo viene pulito e la finestra eliminata, dopodiché sarete riportati al menù di *AppleDisk*.

Come funziona il programma

Il programma è stato realizzato in Basic Applesoft per il personal computer Apple II e, con un drive e stampante.

Può essere suddiviso in tre parti fondamentali: inizializzazione del sistema e menù principale, procedure, sottoprogrammi.

Il menù principale richiama le procedure, che a loro volta richiamano i sottoprogrammi.

100-350 Inizializza il sistema a menù pricipale.

100 Seleziona il video di testo (TEXT): pulizia del video (HOME).

105 Seleziona il modo caratteri in inverso (INVERSE).

110-130 Stampa intestazione.

140 Seleziona il modo caratteri normale (NORMAL).

150 Apre una finestra nel video testo in verticale, partendo dalla 3a riga (esclusa).

155-170 Invia all'utente il messaggio inserito nell'istruzione di PRINT (su video).

180 Aspetta che venga premuto un tasto e lo memorizza in A$ (GET A$): se è diverso dall'asterisco torna a 180.

190-320 Visualizza il menù principale. FLASH seleziona il modo caratteri lampeggianti, VTAB (y) posiziona il cursore alla posizione verticale y, HTAB (x) come VTAB solo in orizzontale.

330 Aspetta che venga premuto un tasto compreso tra 1 e 7.

340-350 A seconda del tasto premuto dall'utente salta ad una procedura.

1050-1100: sottoprogramma M/C. Stampa su video il nome della procedura e chiede di premere M per tornare al menù o C per seguire la procedura scelta. Poi aspetta che vengano premuti i tasti M o C. Infine torna alla procedura che lo ha richiamato.

1050-1070 Pulizia del video e stampa di A$ (contiene il nome della procedura che lo ha richiamato).

1080-1100 Visualizza il messaggio contenuto tra doppi apici, aspetta che venga premuto il tasto M o C e torna alla procedura che lo ha richiamato.

2050-2070: sottoprogramma S/N. Aspetta che venga premuto il tasto S o N e torna al punto di chiamata.

3050-3100: sottoprogramma R N RECORD. Registra sul file sequenziale NRECORD il numero di records (schede) attualmente presenti nell'archivio (file GARE).

3050 Assegna alla variabile DS il carattere di controllo DOS (Disk Operating System) CTRL-D.

3060 Invia il carattere di controllo e informa il DOS che deve effettuare o-

perazioni sul file NRECORD.

3070 Invia il carattere il controllo e informa il DOS che deve effettuare una operazione di scrittura sul file.

3080 Registra sul file la variabile NR che contiene il numero di records (schede) attualmente presenti nel file GARE (l'archivio).

3090 Invia il carattere di controllo e informa il DOS che ha terminato le operazioni sul file precedentemente aperto.

3100 Torna al punto di chiamata.

4050-4100: sottoprogramma I N RECORD. Legge dal file sequenziale NRECORD il numero di records (schede) attualmente presenti nell'archivio (file GARE).

4050 Assegna alla variabile D$ il carattere di controllo del DOS (Disk Operating System) CTRL-D.

4060 Invia il carattere di controllo e informa il DOS che deve effettuare operazioni sul file sequenziale NRECORD.

4070 Invia il carattere di controllo del DOS e lo informa che deve leggere dal file.

4080 Legge il contenuto del file NRECORD e lo assegna alla variabile NR (contiene il numero di records attuali contenuti nel file GARE).

4090 Invia il carattere di controllo e informa il DOS che ha terminato le operazioni sul file.

4100 Torna alla chiamata.

5050-5120: sottoprogramma L RECORD. Legge il record RL dal file GARE e ne memorizza i campi nel vettore SC$.

5060 Invia il carattere di controllo del DOS e lo informa che deve effettuare delle operazioni sul file GARE i cui records sono lunghi 145 bytes (caratteri).

5070 Invia il carattere il controllo e informa il DOS che deve leggere il record RL del file random (R) precedentemente aperto.

5100 Trasferisce i campi del record da buffer nel vettore SC$.

6050-6120: sottoprogramma R RECORD. Registra sul file GARE il record RR contenuto nel vettore SC$.

6070 Invia il carattere di controllo e informa il DOS che deve effettuare un'operazione di scrittura, sul file random precedentemente aperto, del record RR.

6080-6100 Registra il record RR contenuto nel vettore SC$.

7050-7280: sottoprogramma stampa scheda. Stampa la scheda RL sul video, se FS=ø, su carta, se FS=1.

7080 Se FS= 1 seleziona come dispositivo d'uscita la stampante anziché il video.

7120 Scompone il campo contenente la data in giorno, mese, anno e li stampa (vedi descrizione archivio).

7220 Se SC$ (7) = 0 scompone il campo contenente il risultato in ore, minuti, secondi e centesimi di secondo (vedi descrizione archivio).

7260 Se FS=1 ripristina come dispositivo di uscita il video e azzera FS.

Linee 8050-8390: sottoprogramma input scheda.

Chiede all'utente di introdurre i campi del record NS. Se la lunghezza di un campo è superiore a quella predeterminata, viene chiesto di introdurlo nuovamente.

8240 CALL - 958 Cancella tutti i caratteri dopo il cursore.

10060-10110: procedura creazione nuovo schedario. Azzera la variabile NR e la registra nel file NRECORD, per inizializzare l'archivio (0 schede).

10060 Assegna ad AS il nome della procedura.

10070 Richiama il sottoprogramma F/C).

10080 Se O$= M torna al menù (linee 190).

11060-11180: procedura introduzione schede. Chiede all'utente di introdurre una o più schede per ampliare l'archivio ed incrementa ogni volta la variabile NR, che registra a fine procedura.

12060-12370: procedura eliminazione scheda. Elimina una o più schede dall'archivio.

Ogni volta che elimina una scheda, decrementa la variabile NR e compatta i record del file GARE (che contiene le schede).

12115 Se la scheda da eliminare (NS) è l'ultima (NResima) decrementa NR, lo registra sul file NRECORD, e torna al menù principale.

12115 Se la scheda da eliminare è compresa, nel file GARE, tra altre schede, viene eliminata sovrapponendovi le schede (records) successive.

13050-13230: procedura di correzione di una scheda. Registra la scheda (record) corretta su quella errata. La scheda corretta deve essere introdotta nuovamente dall'utente.

14025-14755: ricerca. Dopo aver chiesto all'utente su quale campo e con quale criterio, il programma effettua una ricerca sequenziale sull'archivio e visualizza le schede che possiedono le caratteristiche prescelte.

14040-14110 Menù ricerca n.1 Chiede all'utente su quale campo deve effettuare la ricerca.

14115-14135 Nomi dei campi. Vengono memorizzati nel vettore CR$ per essere utilizzati nel 2° menù ricerca.

14145 Se AS="2", se è si è deciso di effettuare la ricerca sulla data, salta a 14735.

14155 Se AS "6", cioè si effettua la ricerca su una specialità, si salta il secondo menù. Infatti non ha senso confrontare due specialità.

14160-14235 Menù ricerca N.2. Vedi esempio precedente.

14245 Se J=7, cioè si è deciso di effettuare la ricerca sul campo PIAZZAMENTO, si assegna a J il valore 9, indice al quale corrisponde il piazzamento nel vettore SCS.

14265-14730 Ricerca sequenziale.

14740-14755 Input della data in base alla quale bisogna effettuare la ricerca.

15050-15180: procedura stampa su carta. Dopo aver ricercato le schede desiderate con la procedura N. 5, si possono stampare su carta con questa procedura.

16050-16090: procedura fine del programma.

Chiede all'utente se è sicuro di voler terminare, se risponde affermativamente azzera il video e ritorna al menù di *AppleDisk*, altrimenti torna al menù principale.

Sergio Di Nardo
Paolino Di Loreto

AppleDisk

Cinque programmi al mese...più dieci dischetti gratis*

Abbonati. Se ti abboni subito, AppleDisk non costa più nulla. Una confezione di dieci dischi vergini è infatti il regalo di AppleDisk agli abbonati. Dunque...

*per chi si abbona subito, i dischetti vergini e il portadischetti sono compresi nel prezzo dell'abbonamento.

Per sottoscrivere l'abbonamento a AppleDisk e ricevere dieci numeri della rivista con dischetto programmi, più una scatola con dieci dischi vergini, basta compilare e spedire subito questo tagliando a Editronica - Gruppo Editoriale JCE srl - via Ferri 6 - 20092 Cinisello B. (MI)

Sì, mi abbono a AppleDisk.
Speditemi subito (senza alcun aggravio di costi) la confezione con dieci dischetti vergini riservata agli abbonati. Riceverò inoltre dieci numeri di AppleDisk, completi di disco programmi, direttamente al mio indirizzo.

❐ Accludo assegno intestato a Gruppo Editoriale JCE Srl di lire 150.000

❐ Accludo ricevuta di versamento di lire 150.000 sul c/c postale n. **351205** intestato Gruppo Editoriale JCE Srl.

❐ Desidero fattura. Il mio Codice fiscale/Partita Iva è ....

Cognome .... Nome ....

Indirizzo ....

Cap .... Città .... Prov. ....

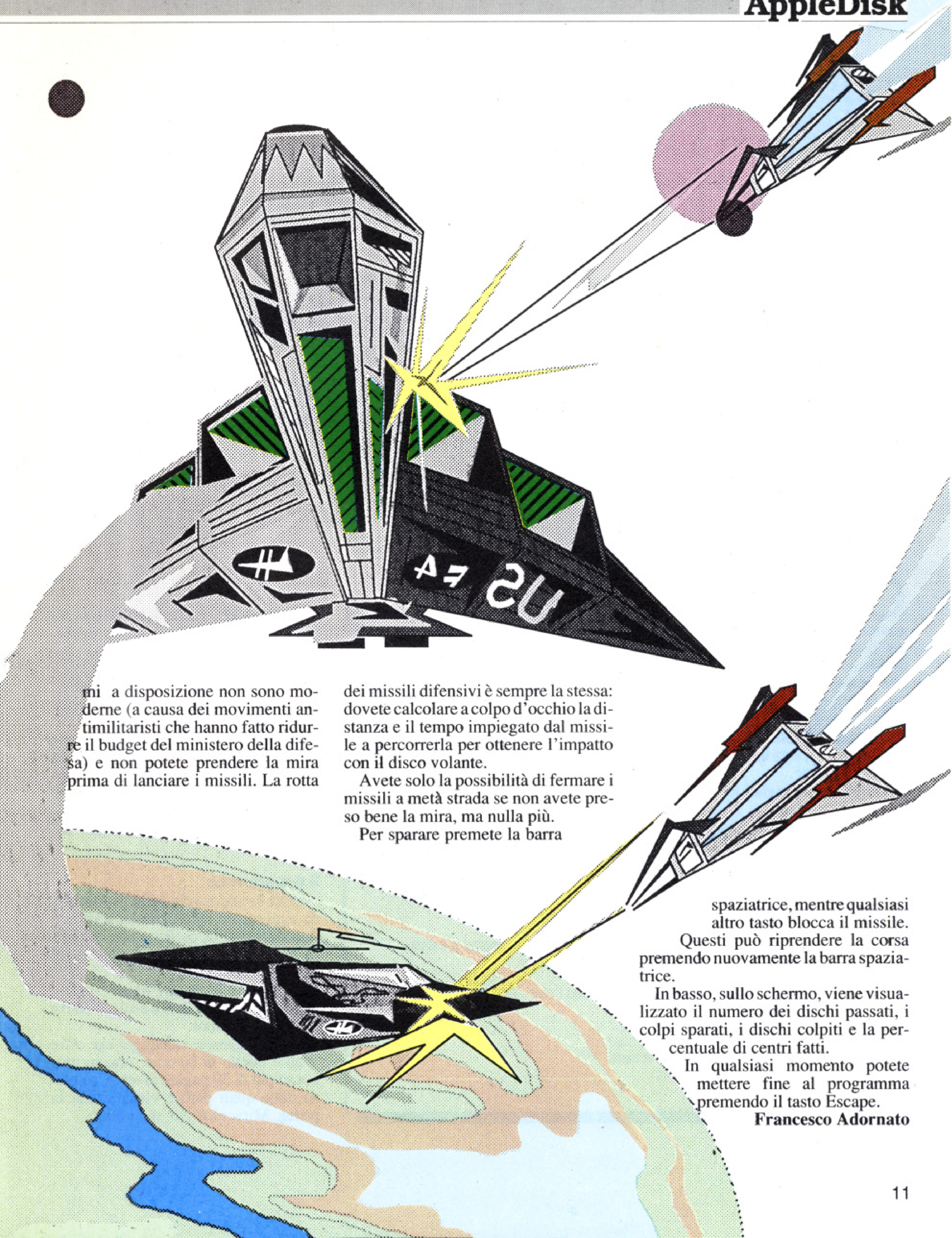
Dischi volanti

Questo gioco spaziale metterà a dura prova i vostri riflessi.
La vostra missione è quella di difendere laTerra da un attacco di alieni.

Ecco un altro gioco spaziale, molto semplice a prima vista, ma che metterà a dura prova la velocità dei vostri riflessi. Si tratta di colpire i dischi volanti che attraversano lo schermo ad altezza variabile, con i missili a vostra disposizione. Riuscirete ad eliminare tutti i dischi volanti che attaccano la terra?

ra e siete al comando di una stazio-
Siete l'unico a difendere la ter-
ne di difesa.

Il vostro compito è quello di distruggere tutti i dischi volanti che passano sopra di voi. Purtroppo le ar-



mi a disposizione non sono moderne (a causa dei movimenti antimilitaristi che hanno fatto ridurre il budget del ministero della difesa) e non potete prendere la mira prima di lanciare i missili. La rotta dei missili difensivi è sempre la stessa: dovete calcolare a colpo d'occhio la distanza e il tempo impiegato dal missile a percorrerla per ottenere l'impatto con il disco volante.

Avete solo la possibilità di fermare i missili a metà strada se non avete preso bene la mira, ma nulla più.

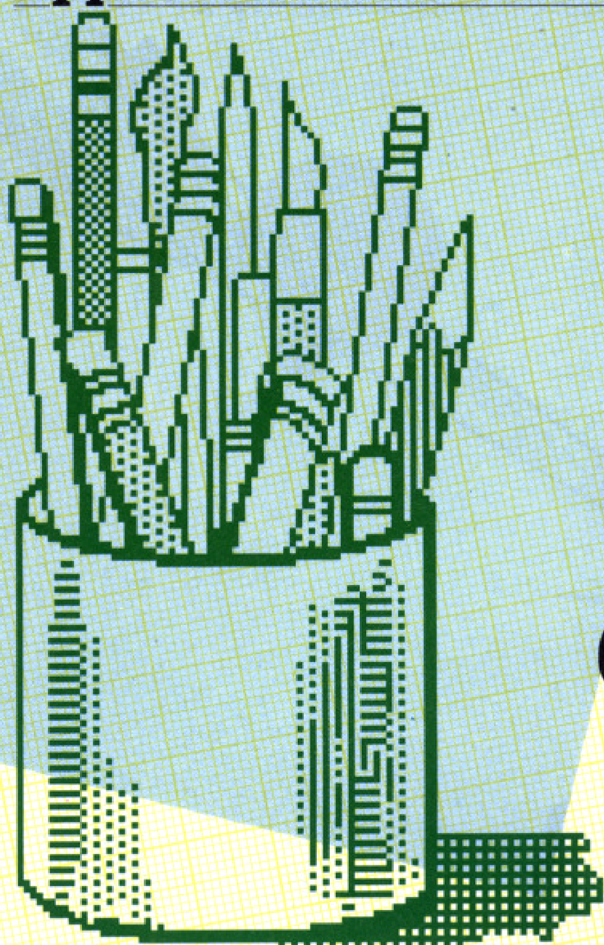
Per sparare premete la barra spaziatrice, mentre qualsiasi altro tasto blocca il missile.

Questi può riprendere la corsa premendo nuovamente la barra spaziatrice.

In basso, sullo schermo, viene visualizzato il numero dei dischi passati, i colpi sparati, i dischi colpiti e la percentuale di centri fatti.

In qualsiasi momento potete mettere fine al programma premendo il tasto Escape.

Francesco Adornato



Abbellite graficamente i vostri programmi con queste potenti utility in grafica a bassa risoluzione.

Disegnare è facile

Chi ha mai programmato o programma ancora con l'Apple II, prima o poi si sarà quasi sicuramente posto il problema di utilizzare la grafica per animazioni, giochi o schemi.

La grafica dell'Apple è organizzata in due pagine in alta risoluzione (280, 192) e due in bassa (40, 48), una delle quali però è coperta dall'inizio dei programmi in basic.

Mentre per le pagine Hi-Res esistono comandi per la gestione di figure vettoriali (vedi DRAW, XDRAW, ROT, SCALE spiegati nel manuale d'uso), per la grafica Lo-Res esistono i soli comandi PLOT, HLIN, VLIN, COLOR=, GR e la funzione SCRN.

Peccato, perché 16 colori sono meglio di 8, anzi, di 6 (anzi, ad essere onesti, soltanto 2) che possiamo avere in alta risoluzione.

Una pagina con 40 x 48 pixel non potrà dare effetti grafici strabilianti, va bene, però qualcosa si può fare; in fondo ogni operazione fatta in bassa risoluzione è 8 volte più veloce o più grande della rispettiva operazione fatta in alta risoluzione. E con un microprocessore quale il 6502 vuol dir molto!

Per questo motivo vi sarà molto utile il programma che vi presentiamo in queste pagine:e cioè Walt che comprende una serie di routines capaci di gestire la pagina in bassa risoluzione, richiamabili da Basic con alcuni comandi.

Il programma

Walt contiene una serie di routines da aggiungere all'Applesoft scritte in Assembler 6502, ovviamente, che permettono di giocherellare facilmente in bassa risoluzione.

Si possono disegnare figure, spostare pagine, sovrapporle e così via fino a creare animazioni.

Il primo grande vantaggio che offre Walt è quello di lavorare non su una, ma su più di 30 pagine diverse, solo una delle quali però è visualizzata: la prima. Miracolo? No, memoria!

Tavola 1. Caratteri speciali

valore del byte	operazione
0..15	traccia un pixel col colore specificato nel byte e spostati a destra.
16	spostati a destra senza tracciare (trasparente).
17	vai a capo (Newline).
18	traccia un pixel col colore impostato da Color= e spostati a destra.
19	spostati in alto senza tracciare.
20	spostati a sinistra senza tracciare.
21	spostati in basso senza tracciare.
22..	fine figura (torna al Basic).

Tavola 2.

numero Kbyte	indirizzo	utilizzo	pagina Walt
0	$0000 - $03FF	riservato	
1	$0400 - $07FF	TEXT E GR	1
2..7	$0800 - §1FFF	prog. BASIC	(fine prog.)..7
8..15	$2000 - $3FFF	HGR	8..15
16	$4000 - §43FF	OBJ WALT	
17..36	$93FF	figure walt	17..36
37..	$9400..	DOS e Applesoft	

Un pixel o una figura possono essere tracciati sulle pagine che si vuole e inoltre, altro vantaggio di Walt, i pixel non vanno da 0 a 39 (x) e da 0 a 47 (y), ma da 100 a 139 e da 100 a 147. Tutto quello che viene disegnato fuori da questa finestra **non** viene visualizzato.

Prima di utilizzare Walt, occorre caricarlo con

BLOAD OBJ WALT

e bisogna effettuare alcune inizializzazioni: è necessario inserire nel proprio programma Basic le assegnazioni degli indirizzi delle routines di Walt alle rispettive variabili. Basta digitare

LOAD VAR WALT

e, chiamando LIST, apparirà:

I comandi

CALL MOVE%, a, b : Sposta la pagina (a) nella pagina (b) cancellando quest'ultima.

CALL PICT%,n, x, y, p : Disegna la figura (n) a partire dalle coordinate (x), (y) della pagina (p).

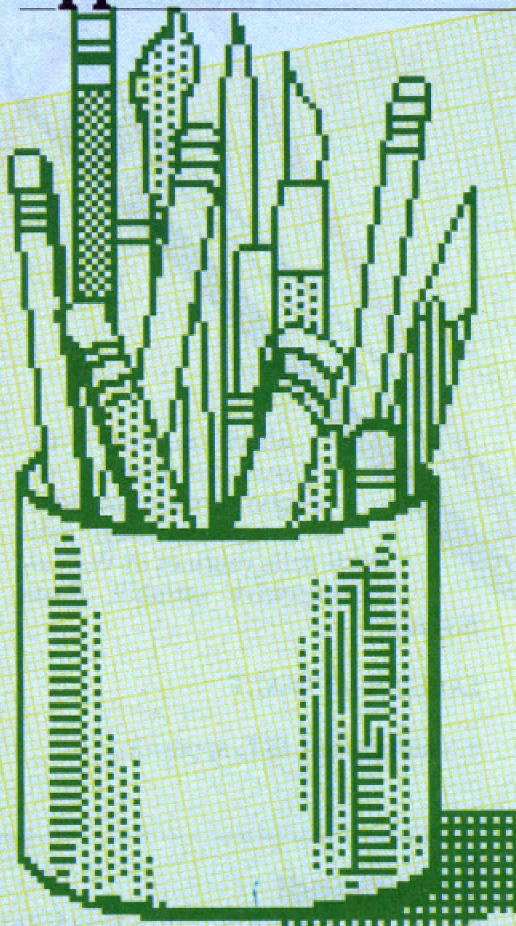
Più avanti è spiegato come le figure vengono implementate in memoria.

CALL PXEL%, x, y,: Disegna un pixel alle coordinate (x), (y) della pagina (p) col valore impostato precedentemente dal comando Applesoft COLOR=.

CALL ADD%, a, b: Copia i punti NON NERI della pagina (a) sopra la pagina (b).

CALL CMP%, x, y, a, b: Confronta il punto (x, y) della pagina (a) con lo stesso punto della pagina (b).

In uscita abbiamo nella locazione $06 (PEEK (6)) il colore del punto della pagina (a), nella locazione $07 (PEEK (7)) il colore di quello della pagina (b) e nella locazione $09 (PEEK (9)) la differenza tra i due valori.



```
1000 REM
1010 REM INDIRIZZI WALT
1020 REM
1030 :
1040 WALT% = 16384
1050 MOVE% = WALT% + 0
1060 PICT% = WALT% + 105
1070 PXEL% = WALT% + 332
1080 ADD% = WALT% + 385
1090 CMP% = WALT% + 497
1100 : REM
```

quindi potete scrivere il vostro programma.

Un consiglio: aggiungete un PRINT CHR$(4); "BLOAD OBJ WALT" in linea 1035 solo quando il vostro programma è finito e funzionante, pronto per il Lock.

Le figure

Le figure sono memorizzate assegnando semplicemente ad un singolo byte il codice del colore di un pixel, anche perché oltre ai 16 colori sono stati implementati alcuni caratteri cosiddetti speciali secondo la **tavola 1**.

Queste figure così definite sono raggruppate in una tavola organizzata con la stessa logica con la quale si organizzano le tavole delle figure vettoriali in alta risoluzione come è visibile nella tavola 3.

Tavola 3.

inizio tavola
numero (n) di figure della tavola

directory
indirizzo relativo fig. 1 (L)
indirizzo relativo fig. 1 (H)
indirizzo relativo fig. 2 (L)
indirizzo relativo fig. 1 (H)

indirizzo relativo fig. n (L)
indirizzo relativo fig. n (H)

figura 1
codici della fig. 1

figura 2
codici della fig. 2

figura n
codici della fig. n

Alla fine bisogna specificare l'indirizzo dell'inizio della tavola nelle locazioni 250 e 251. Ma niente paura! A fare tutto questo ci pensa un apposito programmino che si chiama BAS WALTEDIT. A tutti quelli che stavano

per chiudere la rivista sconsolati per non aver capito quasi niente, dedichiamo questo programma, scritto interamente in Basic, e facile da usare. Il programma va solo caricato e, per il momento, non va messo in esecuzione.

Quindi dalla linea 4000 in poi potete sbizzarrirvi a suon di Data a creare figure seguendo le seguenti indicazioni:

"BEGIN" - inizia figura
"END;" - fine figura
"END." - fine tavola

"A".."P" - colore 0.. colore 15
"." - trasparente
"^" - spostati in alto
"V" - spostati in basso
"<" - spostati a sinistra
"-" - colore impostato da COLOR=.

Ecco un esempio:

```
LOAD BAS WALTEDIT
4000 DATA "BEGIN"
4010 DATA "—"
4020 DATA ".F-F"
4030 DATA "—F"
4040  DATA "-F-F"
4050 DATA "—F"
4060 DATA ".FFF"
4070 DATA "END;"
4080 REM
9000 DATA "END."
RUN
```

Se quando vi verrà chiesto "Vuoi vedere i disegni?" rispondete di Sì, verrà visualizzato solo un pasticcio! Abbiamo infatti disegnato nero su nero (con un po' di grigio). Provate ora a modificare la linea 3120 da

```
3120 GR: HOME
```

con

```
3120 GR: HOME: COLOR= 15
```

e date il RUN.

Questa volta WALTEDIT non rilegge il codice oggetto OBJ WALT (perché lo ha già letto prima) e disegnerà una A in bianco con ombreggiatura grigia. Un altro esempio potete trovarlo nel file PIC PINGUINI: fate il LOAD PIC PINGUINI e guardate il listato dalla linea 4000 in poi.

Le pagine

Spiegare l'organizzazione delle pagine di Walt è un po' ripetere l'organizzazione della memoria dell'Apple.

Ogni pagina è lunga un Kbyte, quindi si dispongono seguendo la **tavola 2**.

OBJ WALT è memorizzato a partire dalla locazione $4000 cioè sopra HGR2 (pagina 16 del Walt). WALTEDIT, se non specificato, pone le figure a partire dalla pagina diciassette del Walt. Vedete quindi, in base alla lunghezza della tavola, alla lunghezza del programma e al numero di pagine che vi servono, dove collocare queste ultime, facendo attenzione a non distruggere le figure, il Dos, il programma Basic o addirittura lo stesso Walt.

Demo

Nel dischetto sono memorizzati alcuni programmi dimostrativi: BAS DEMO1 e BAS DEMO2 mostrano alcune semplici animazioni, BAS DEMO0 dà alcuni esempi di ciò che si può fare col Walt. BAS CORSA è un giochino, scritto sempre in Basic, di facile esecuzione. Tutto questo è solo un esempio di ciò che si può fare con Walt in un paio d'ore.

Walt&S

WALT&S sta per Walt & Sound. Tutto quanto appena visto, se fatto con Walt & Sound (cioè OBJ WALT&S e VAR WALT&S) è anche sonoro (ovviamente i suoni sono a caso).

Conclusioni

Walt, per la lentezza del Basic dal quale viene usato, purtroppo non è molto adatto ai videogames, comunque può essere usato per scritte, brevi animazioni e cose di questo tipo con buoni risultati.

Walt sarà sicuramente d'aiuto alla creatività di chi, fino ad ora, non ha avuto i mezzi (cioè: programmi) per applicarla se non quello di cambiare computer mandando in pensione il glorioso maestro Apple II.

Marco Busetto

Copiare con FIDucia

FID è un'utility preziosa per i lettori di AppleDisk: permette di effettuare con facilità una copia di sicurezza dell'AppleDisk acquistato, trasferendo i programmi in esso contenuti su altri dischetti. Per effettuare la copia dell'intero disco o di alcuni programmi, il disco destinazione deve essere formattato in precedenza seguendo le seguenti istruzioni.

• Come formattare un dischetto in DOS 3.3:

- Inserite l'AppleDisk nel drive 1 e accendete il computer.
- Dal menù principale scegliete Fine e premete Return.
- Togliete il dischetto dal drive e sostituitelo con un dischetto nuovo.
- Scrivete NEW e premete Return.
- Scrivete INIT HELLO e premete Return.
- Il drive girerà per circa 30 secondi e al termine avrete il dischetto formattato in DOS 3.3 e autobooting.

Ora, con il programma FID, potrete trasferire su questo dischetto i programmi che desiderate.

• Opzioni del programma: 1. Copia File; 2. Catalog; 3. Settori vuoti; 4. Sblocca file; 5. Blocca file; 6 .Elimina file; 7. Reset slot e drive (Reset); 8. Verifica file (Verify); 9. Quit (fine programma).

Modifiche Fid

• Schedario Atletica.
Trasferite il file Schedario Atletica.

• Labirinto.
Trasferite il file Labirinto, Shapes, Tune.

• Trucchi Hi-Res.
Trasferite i file Hires.Tricks, Hires.Demo, Demo.Pic1, Demo.Pic2.

• Dischi Volanti.
Trasferite il file Dischi Volanti.

• Notazione Scientifica.
Trasferite il file Notazione Scientifica.

• Grafica GR.
Trasferite il file Explorer 1.0.

• Disco detective.
Trasferite il file Walt.

Queste potenti e flessibili routine possono essere usate facilmente in ambiente Basic o in linguaggio macchina per conferire vivacità ai vostri programmi.

Il trucco c'è ma non si vede

Hi-Res Tricks, trucchi in alta risoluzione, è un pacchetto di 15 routine in linguaggio macchina che permette di manipolare le immagini sullo schermo della grafica ad alta risoluzione dell'Apple.

Potete creare videate in reverse, scorrimento orizzontale e verticale, immagini speculari (**figura 1**), videate capovolte e altro ancora. Hi-Res Tricks può dare ai vostri programmi un maggiore impatto visivo, e per usare le routine non è necessario conoscere il linguaggio macchina.

Le routine

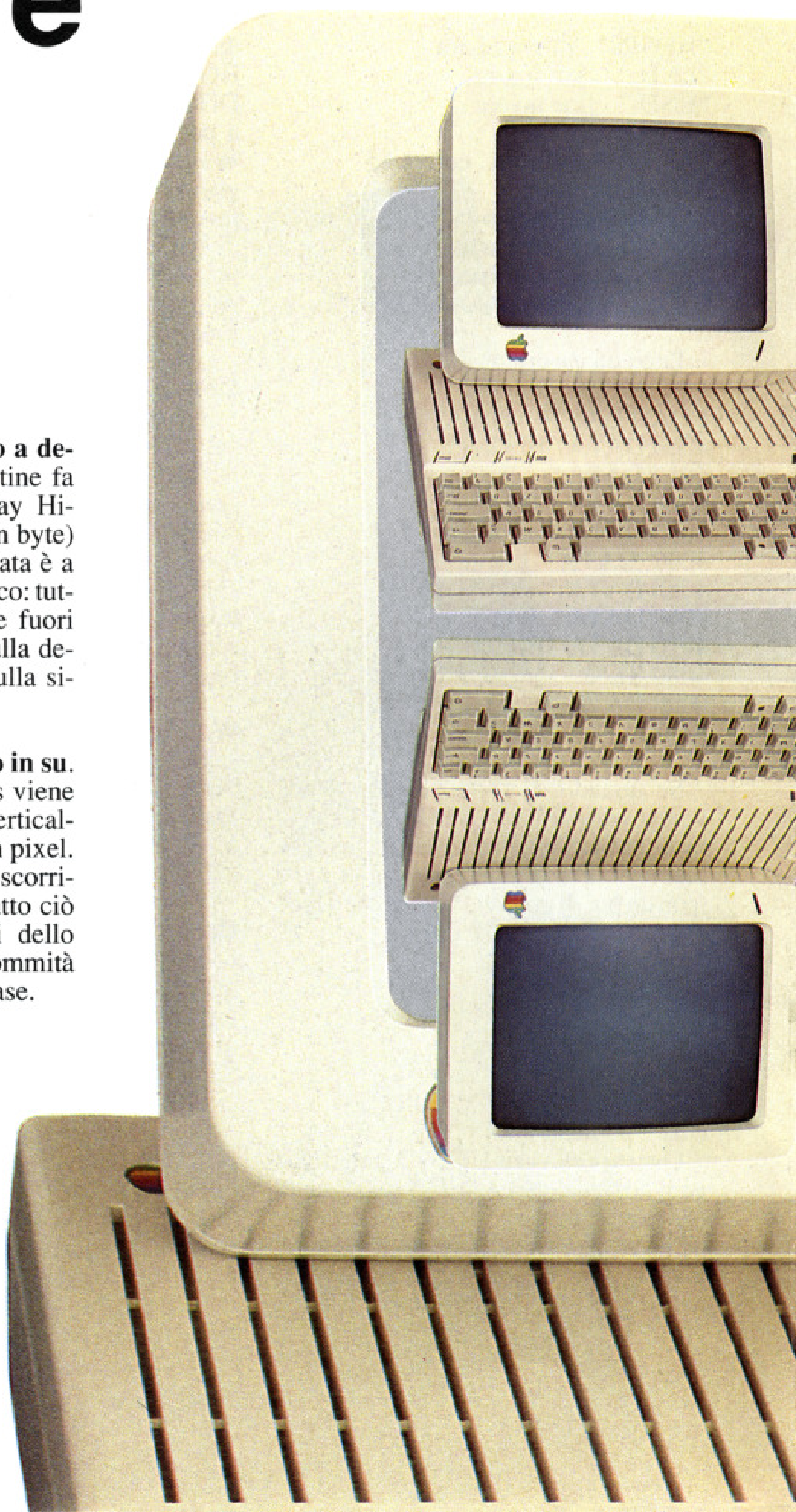
Esaminiamo le 15 routine, una alla volta:

• **Creazione negativa**. Questa routine inverte l'intero schermo Hi-Res. Tutti i pixel (puntini) che erano accesi vengono spenti e tutti i pixel che erano spenti vengono accesi.

• **Spostamento a sinistra**. Questa routine fa scorrere il display in alta risoluzione di 7 pixel (un byte) a sinistra. La videata è a scorrimento ciclico: tutto ciò che scorre fuori dello schermo sulla sinistra ricompare sulla destra.

• **Spostamento a destra**. Questa routine fa scorrere il display Hi-Res di 7 pixel (un byte) a destra. La videata è a scorrimento ciclico: tutto ciò che scorre fuori dello schermo sulla destra ricompare sulla sinistra.

• **Spostamento in su**. Il display Hi-Res viene fatto scorrere verticalmente in su di un pixel. La videata è a scorrimento ciclico: tutto ciò che scorre fuori dello schermo alla sommità ricompare alla base.

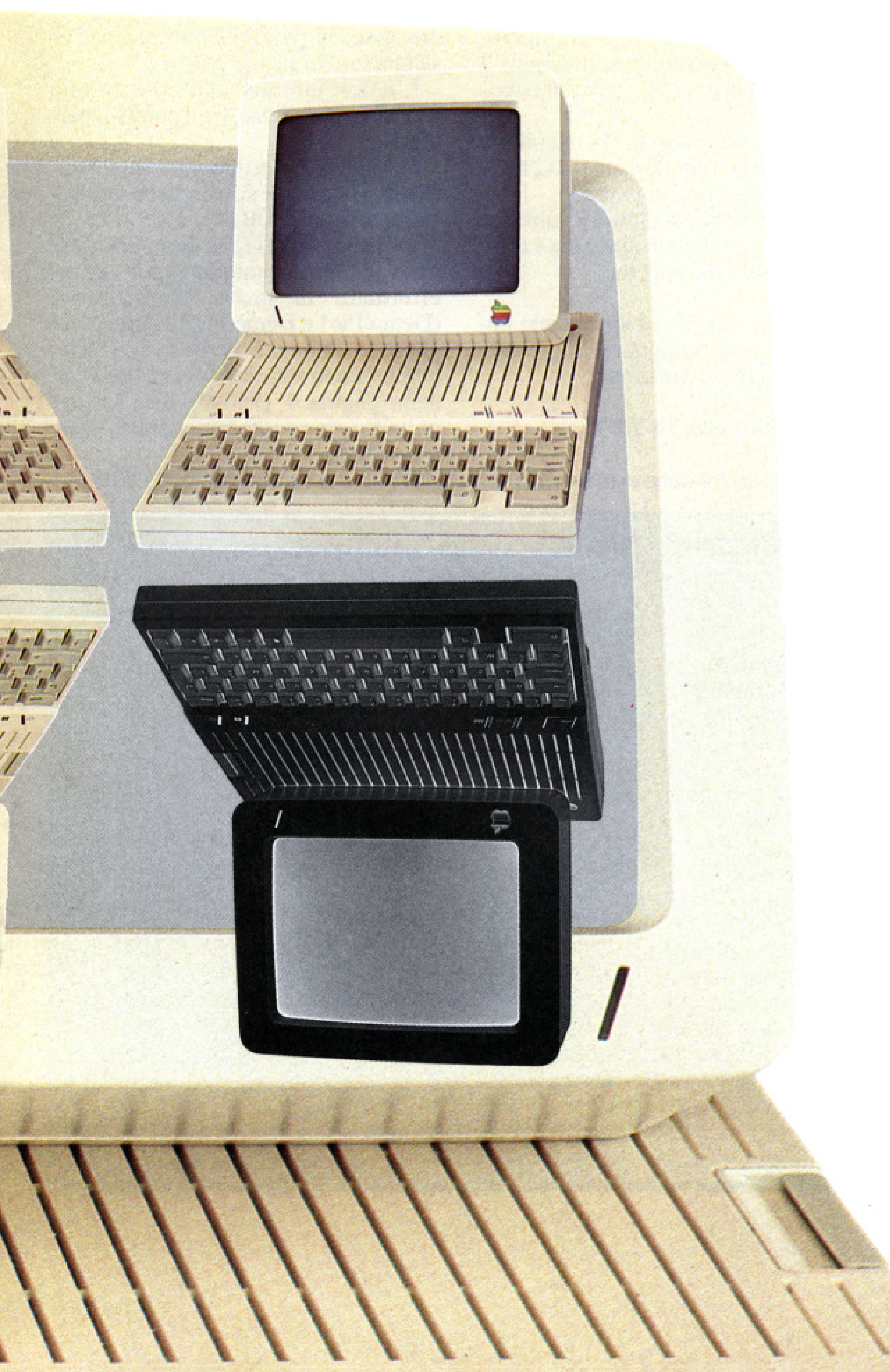


• **Spostamento in giù**. Questa routine fa scorrere il display Hi-Res verticalmente in giù di un pixel.

La videata è a scorrimento ciclico: tutto ciò che scorre fuori dello schermo alla base ricompare alla sommità.

• **Inversione sinistra-destra**. Scambiando i lati sinistro e destro della videata l'immagine viene ribaltata in senso orizzontale.

• **Inversione alto-basso**. Scambiando la sommità e la base della videata l'immagine viene ribaltata in senso verticale.



• **Pagina 1 più pagina 2**. Questa routine prende il contenuto della pagina 2 Hi-Res e lo aggiunge a qualunque cosa si trovi sulla pagina 1. L'effetto è come una sovrapposizione trasparente.

• **Pagina 1 meno pagina 2**. Questa routine sottrae il contenuto della pagina 2 Hi-Res dalla pagina 1. Ciò cancella in pratica alcune parti della pagina 1, a seconda di ciò che si trova sulla pagina 2.

• **Pagina 2 trasferita in pagina 1**. Il contenuto della pagina 2 viene copiato sulla pagina 1, cancellando nell'operazione tutto ciò che c'era a pagina 1.

• **Pagina 1 trasferita a pagina 2**. Questa routine copia il contenuto della pagina 1 sulla pagina 2, cancellando nell'operazione tutto ciò che c'era a pagina 2.

• **Pagina 2 trasferita in pagina 1 (speciale)**. Questa routine è speciale perché vengono trasferite sulla pagina 1 solo alcune parti della pagina 2. Questo rende possibile avere tante varianti di un display quante se ne possono far stare su una pagina di grafica Hi-Res. Si può allestire la pagina 1 come sfondo generale, poi si possono trasferire a ripetizione sulla pagina 1 parti della pagina 2 per avere display differenti.

• **Spostamento a sinistra di due pagine**. Entrambe le pagine Hi-Res vengono fatte scorrere a sinistra come se fossero una sola unità. L'effetto è quello dello scorrimento di una videata larga il doppio.

• **Spostamento a destra di due pagine**. Questa routine fa scorrere a destra entrambe le pagine Hi-Res.

• **Spostamento in su di due pagine**.

Entıambe le pagine Hi-Res vengono fatte scorrere in su come se fossero una sola unità. L'effetto è quello dello scorrimento di una videata alta il doppio.

• **Spostamento in giù di due pagine**. Questa routine fa scorrere in giù entrambe le pagine Hi-Res.

Come usare il programma

Hi-Res Tricks viene caricato in memoria all'indirizzo esadecimale $6000 e la sua lunghezza è $609.

Ognuna delle routine di Hi-Res Tricks può essere usata separatamente e ognuna può essere caricata in qualsiasi punto della memoria. Questo le rende molto flessibili.

Per usare una routine in ambiente Basic basta caricarla con BLOAD in una sezione non utilizzata della memoria e richiamarla con CALL, usando l'indirizzo al quale l'avete caricata. Hires.Tricks contiene tutte e 15 le routine come unico grande file, ma lo potete suddividere in conformità con le vostre esigenze.

Hires.Demo carica tutte le routine in memoria e richiama per ogni dimostrazione quella appropriata. La **tavola1** indica gli indirizzi di inizio delle routine e la loro lunghezza.

Per usare le routine con il linguaggio macchina basta saltare con JSR alla locazione nella quale avete memorizzato la routine. Dovete salvare i registri X e Y se contengono dati importanti; le routine cambieranno infatti i valori che vi sono immagazzinati.

Come indicato nella **tavola 1**, le prime sette routine (da Creazione di negativo a Inversione alto-basso) sono predisposte per l'azione sulla pagina 1 tuttavia è possibile modificarle in modo che operino sulla pagina 2 (vedere la sezione Modifiche in questo articolo).

La routine denominata Pagina 2 trasferita in pagina 1 (speciale) richiede qualche spiegazione supplementare. Dato che sulla pagina 1 vengono trasferite solo parti della pagina 2 si devono predisporre anzitutto tre locazioni di memoria, prima di richiamare la routine.

1. $658D (decimale 25997) stabilisce la colonna orizzontale di pagina 2 dove il programma ha inizio il trasferimento.
2. $6605 (decimale 26117) stabilisce la colonna orizzontale di pagina 2 dove mette fine al trasferimento.
3. $6599 (decimale 26009) stabilisce la colonna orizzontale di pagina 1 dove ha inizio il trasferimento.

Queste locazioni possono essere impostate da un programma in linguaggio macchina con l'istruzione:

LDA #$XX, STA $YYYY

o in Basic scrivendone in memoria i valori con istruzioni POKE.

Sono disponibili quaranta colonne, da 0 a 39. Se il numero che scrivete con POKE non è in questa fascia la routine non funzionerà a dovere e può darsi che sovrascriva qualche parte della memoria.

Non c'è arresto orizzontale per la pagina 1. Siano 2, o 5, o 35 le colonne che intendete trasferire dalla pagina 2 dev'essere disponibile sulla pagina 1 esattamente lo stesso spazio.

L'unica informazione che dovete fornire per la pagina 1 è il punto di partenza.

Per eseguire il programma dimostrativo dovrete avere sul dischetto due vostre immagini Hi-Res.

In questo *AppleDisk* sono già state inserite due immagini per poter effettuare la prova dimostrativa (Demo.Pic1 e Demo.Pic2). Potete scegliere le immagini Hi-Res che volete, purché siano state salvate con BSAVE dalla pagina 1 ($2000). Per salvare un'immagine nel formato giusto usate il comando:

BSAVE immagine,A$2000,L$2000

Tavola 1. Indirizzi di chiamata delle routine

Routine	Locazione iniziale Esadecimale	Locazione iniziale Decimale	Lunghezza Esadecimale	Lunghezza Decimale
Creazione di negativo	$6000	24576	$1C	28
Spostamento a sinistra	$601C	24604	$4E	78
Spostamento a destra	$606A	24682	$4E	78
Spostamento in su	$60B8	24760	$58	88
Spostamento in giù	$6110	24848	$58	88
Inversione sinistra-destra	$6168	24936	$8E	142
Inversione alto-basso	$61F6	25078	$A8	168
Pagina 1 + pagina 2	$629E	25246	$24	36
Pagina 1 - pagina 2	$62C2	25282	$26	38
Pagina 1 ->pagina 2	$62E8	25320	$22	34
Pagina 2 ->pagina 1	$630A	25354	$22	34
Spost. sin. di due pagine	$632C	25388	$7C	124
Spost. dest. di due pagine	$63A8	25512	$7C	124
Spost. in su di due pagine	$6424	25636	$B4	180
Spost. in giù di due pagine	$64D8	25816	$B4	180
Pag. 2 — >pag. 1 speciale	$658C	25996	$7D	125

Nota: queste locazioni sono valide solo se le routine vengono caricate in memoria al loro indirizzo originale.

Quando eseguite la dimostrazione scrivendo RUN HIRES.DEMO vi vengono chiesti i nomi di file delle due immagini che volete usare. Introducete i loro nomi e il programma le manipolerà usando molte delle routine di Hi-Res Tricks.

Come funziona

Tutte le routine utilizzano la pagina zero per la memorizzazione temporanea. Questo viene fatto per due ragioni: primo, permette la rilocabilità; secondo, la pagina zero è più veloce e molto più facile da usare delle altre aree. In tutti i programmi il registro Y viene usato per indicizzare indirettamente l'indirizzo base memorizzato nella pagina zero in modo che l'indirizzo effettivo sia indicizzato orizzontalmente attraverso la pagina Hi-Res. Invece il registro Y viene usato per l'indicizzazione verticale in giù lungo la pagina.

• Creazione di negativo

Dopo aver memorizzato il primo indirizzo nella pagina zero e impostato su zero l'indice nel registro Y la routine riceve il valore di una locazione nell'area Hi-Res e poi lo complementa. In altre parole se un bit era 0 viene messo a 1, se era 1 viene messo a 0. A far questo provvede l'OR esclusivo (EOR #$FF).

Il valore complementato viene poi riportato alla sua locazione originale, l'indice viene incrementato e viene complementata la locazione seguente. Questo procedimento è ripetuto fino a che l'indice ha percorso tutti i valori da $00 a $FF.

Poi viene incrementato il byte alto dell'indirizzo, e il procedimento è ripetuto fino a che viene raggiunta la fine della pagina. A fine pagina il byte alto ha un valore di $40 o più. Questo test viene utilizzato in tutte le routine per determinare quando ciascuna di esse deve avere termine. Poi la routine torna al programma chiamante.

• Spostamento a sinistra e a destra

Lo spostamento a sinistra comincia salvando il primo byte di una riga orizzontale dello schermo. Poi ogni byte è spostato di una posizione a sinistra. Infine il primo byte salvato viene messo nell'ultimo byte. Questo crea l'effetto dello scorrimento a sinistra della videata.

Per indicizzare orizzontalmente lo schermo viene incrementato il registro Y, che indicizza indirettamente l'indirizzo base nelle locazioni di pagina zero $FA e $FB. Dopo essersi spostato da un capo all'altro di una riga il programma passa alla riga seguente. Per andare alla riga seguente viene addizionato il valore $04 al byte alto dell'indirizzo base. Poi il byte alto viene controllato per vedere se è stata raggiunta la fine dell'area Hi-Res; in caso negativo la routine sposta quella riga. Quindi viene sottratto il valore $1F80 dall'indirizzo base per tornare nella pagina Hi-Res. Viene usato questo valore perché la pagina Hi-Res non è immagazzinata in memoria sequenzialmente, ma in otto sezioni interallacciate. Il registro X tiene nota della sezione della pagina Hi-Res sulla quale si sta intervenendo. Per esempio dopo il completamento della prima sezione l'indirizzo base in $FA e $FB è $4000. Sottraendo $1F80 si ha un indirizzo base di $2080, che è l'indirizzo della sezione seguente.

Completata la prima sezione, l'indirizzo base in $FA e $FB è adesso $4380. Sottraendo $1F80 si ha ancora $2400, ma adesso il valore del registro X è zero di modo che il programma sottrae un ulteriore $03D8, che dà $2028. Questo è l'indirizzo dell'inizio della seconda sezione. Un analogo procedimento porta la routine alla terza sezione. Completata la terza sezione e fatte le sottrazioni il valore del byte basso dell'indirizzo base è $78. Dato che questa è la sola volta che valore viene raggiunto è utilizzato per determinare la fine della routine.

Le procedure della routine di Spostamento a destra sono simili a quelle della routine di Spostamento a sinistra ma in ordine inverso. Le addizioni e le sottrazioni di $04 sono le stesse e si scende ancora lungo lo schermo. La differenza in questo caso è che invece di cominciare da $00 e andare a $27 il registro Y adesso va da $27 a $00. Il valore dell'ultimo byte è messo ora nella prima locazione per creare l'effetto di scorrimento.

• Spostamento in su e in giù

Per quanto queste routine siano analoghe a quelle di spostamento laterale, hanno bisogno di un ulteriore indirizzo base, e il registro Y viene impostato per primo (cioè prima del registro X). Questo permette alla routine di indicizzare gli indirizzi prima verticalmente e poi orizzontalmente.

Il byte situato più in alto in una colonna verticale viene temporaneamente salvato. Poi ogni byte è spostato in su di una riga e la parola salvata viene messa alla base della colonna.

Gli indirizzi di riga sono calcolati come negli spostamenti laterali, ma vengono ricalcolati completamente per ogni spostamento orizzontale, rallentando un po' il programma. Dato che ogni esecuzione sposta la videata di un solo pixel lo spostamento in su o in giù è più lento dello spostamento laterale. È però possibile accelerare il programma se vi interessa più la velocità della rilocabilità.

• Inversione sinistra-destra

I calcoli degli indirizzi nella routine di Inversione sinistra-destra sono gli stessi della routine di spostamento a sinistra perché il programma procede in giù sullo schermo. Anche in questo caso il registro Y è l'indice orizzontale; il programma prende però un valore da un indirizzo su una riga e lo mette in un altro a una certa distanza, per cui occorrono due indici. Questi sono memorizzati nella pagina zero e caricati nel registro Y secondo necessità. Un indice va da $00 a $14 e l'altro va da $27 a $15. A mano a mano che il programma procede il primo indice viene incrementato e il secondo decrementato.

Per primi vengono scambiati i byte delle colonne orizzontali $00 e $27, poi quelli di $01 e $26 e così via. Questo mette il primo byte nell'ultima locazione e l'ultimo nella prima. Poi il secondo va nella penultima, il penultimo nella seconda, e via dicendo fino a che

vengono scambiate $14 e $15. Dopo il completamento della prima riga orizzontale la routine scende sullo schermo alla riga seguente.

Il semplice spostamento di un byte da una locazione di schermo a un'altra non rovescerà l'intera immagine. Dev'essere invertito anche l'ordine dei bit all'interno del byte. I bit vengono invertiti con poca difficoltà, caricando un byte nell'accumulatore, ruotando l'accumulatore a destra (ROR) e poi ruotando una locazione di memoria a sinistra (ROL).

Facendo il ROR dell'accumulatore il bit all'estrema destra va nel bit di carry, e i restanti bit vengono spostati di una posizione a destra. Quando si fa il ROL della locazione di memoria tutti i bit vengono spostati di una locazione a sinistra e il bit di carry va nella prima locazione. Purché la locazione di memoria sia prima azzerata la sequenza dei bit in memoria sarà invertita rispetto a quella che era in origine nell'accumulatore.È necessario un altro passo. Dato che il display è creato dai sette bit più a destra di una parola (i bit dell'immagine) l'operazione di ROR e ROL viene fatta solo sette volte. L'ottavo bit è il bit di colore e dev'essere conservato, altrimenti l'arancione diventa verde e il blu diventa viola.

L'accumulatore viene ruotato a destra ancora una volta per mettere l'ultimo bit nel bit di carry, ma questa volta, invece di ruotare a sinistra la locazione di memoria, si fa solo il test del bit di carry. Se è a 1, a indicare che l'ottavo bit era 1, viene fatto l'OR del valore in memoria con $80, il che mette nuovamente a 1 l'ottavo bit. Se il carry è a 0 quella parte del programma viene saltata, e l'ottavo bit resta così 0.

• Inversione alto-basso

Come i programmi di Spostamento in su e di Spostamento in giù questa routine avanza simultaneamente in su e in giù sulla pagina. Come l'Inversione sinistra-destra scambia i valori di due locazioni corrispondenti, ma non ha bisogno di invertire l'ordine dei bit delle parole. Questa routine scambia prima le parole nei terzi superiore e inferiore della pagina, poi scambia le parole della sezione mediana. Inoltre, dato che opera orizzontalmente attraverso la pagina, non occorrono locazioni indice separate. Occorrono invece locazioni aggiuntive di pagina zero per memorizzare altri indirizzi base.

• Pagina 1 più pagina 2

Ci sono due metodi per eseguire l'operazione, a seconda del colore di fondo della pagina 2. Dato che la maggior parte dei grafici e dei diagrammi ha lo sfondo nero è così che lo vedrete quando eseguite Hires.Demo. Lo si può modificare molto facilmente per le immagini e per i grafici in modo che sia bianco o colorato.Per combinare le due pagine si devono combinare i bit che formano le pagine. Se un bit è 0 in un byte di pagina 1 lo dobbiamo mantenere 0. Se è 1 lo dobbiamo mantenere 1.

Se tutti i bit di uno dei byte sono 0, facendone l'AND con un altro byte tutti i bit di quel byte diventeranno anch'essi 0. Per ottenere uno sfondo nero tutti i bit dello sfondo devono essere 0. Quindi per combinare bit che sono 0 nella pagina 2 con bit che sono 1 nella pagina 1 si deve fare l'OR dei byte. Così si mette a 1 un bit se è 1 in uno dei byte o in entrambi.

Per avere uno sfondo bianco o colorato l'effetto è rovesciato. Supponendo che ci sia uno sfondo bianco a pagina 2 tutti i bit saranno 1.

Facendo l'OR del byte di pagina 2 con un byte di pagina 1 andranno a 1 tutti i bit compresi quelli, se ce ne sono, che dovrebbero essere 0. In questo caso si deve fare l'AND dei bit, in modo che se un bit è 0 nella pagina 1 lo stesso bit sarà 0 nel byte combinato. Nell'operazione AND un bit è 1 solo se lo stesso bit è 1 nella prima parola e nella seconda.Come presentata, questa routine funziona solo se il colore dello sfondo di pagina 2 è il nero. Per cambiarlo e avere uno sfondo bianco o colorato cambiate in $31 il byte della locazione $62B0. Così quel passo viene cambiato da OR a AND.

• Pagina 1 meno pagina 2

Questa routine rovescia il processo della routine pagina 1 più pagina 2. La locazione $62D4 contiene un OR esclusivo (EOR). Facendo l'EOR del byte da pagina 2 i bit vengono complementati o invertiti: se un bit è 1 diventa 0 mentre se è 0 diventa 1. Poi il byte complementato viene combinato con l'equivalente parola della pagina 1.

Complementando il byte abbiamo in pratica rovesciato il colore. Se in origine era nero (tutti i bit 0) adesso sarà bianco (tutti i bit 1). Così, seguendo la stessa logica vista sopra, si deve fare l'AND dei byte.Per cambiare questo programma in modo che gestisca uno sfondo bianco o colorato a pagina 2 si deve cambiare in OR ($11) l'istruzione della locazione $62D6 perché lo sfondo bianco sarà in pratica cambiato in nero dall'operazione di EOR.

Modifiche

Le routine di Hi-Res Tricks possono essere modificate per creare più effetti in alta risoluzione. Per esempio potete spostare il display a sinistra, a destra, in su e in giù senza effetto di scorrimento. Potete anche creare immagini speculari dei vostri display.

Altre facili modifiche faranno agire le routine sulla pagina 2. Modifiche più complesse permetteranno alle routine di operare solo su specifiche sezioni delle pagine.

Gli indirizzi esadecimali e le istruzioni POKE che seguono sono validi solo se le routine vengono caricate in memoria al loro indirizzo originale. Se per qualsiasi ragione le avrete rilocate dovrete ritoccare gli indirizzi.

Prima di tentare una qualsiasi di queste modifiche assicuratevi di avere una copia della versione originale a disposizione come sicurezza.

• Niente scorrimento

Lo spostamento della pagina Hi-Res a sinistra, a destra, in su o in giù senza scorrimento cancella il display quando ha superato l'orlo della pagina. La parte del display che scorre fuori dello schermo è perduta per sempre; non c'è modo di riportarla indietro. Perciò assicuratevi di avere messo al sicuro la vostra grafica Hi-Res salvandola sul

dischetto se la volete usare ancora in futuro!

Per modificare la routine di Spostamento a sinistra in modo che non scorra, cambiate le locazioni $6039 e $603A in istruzioni NOP ($EA) in ambiente Monitor o inserite l'istruzione POKE 24633,234 : POKE 24634,234 in Basic. Comunque, tutto ciò che si trova nell'ultima colonna verticale sulla destra sarà poi duplicato ogni volta che il programma verrà eseguito.

Si possono fare analoghe modifiche per eseguire spostamenti a destra, in su e in giù senza scorrimento.

• Immagini speculari

Cambiando il contenuto delle locazioni $61B6 e $61B7 in $EA (POKE 25014,234 : POKE 25015,234) si duplicherà il lato sinistro della pagina sul lato destro, ma non si trasferirà la destra sulla sinistra. Il lato sinistro resta immutato e si produce un effetto speculare nel quale il lato sinistro è riflesso sulla destra come in uno specchio.

Analogamente, cambiando il contenuto delle locazioni $61B0 e $61B1 in $EA (POKE 25008,234 : POKE 25009,234) si otterrà il riflesso speculare del lato destro sulla sinistra.

• Operazioni in pagina due

Per operare i programmi sulla pagina 2 basta cambiare come indicato qui sotto le locazioni degli indirizzi base e il test per la fine della pagina:

- *Creazione di negativo*. Cambiare la locazione $6005 in $40 e cambiare $6018 in $60 (usando l'istruzione POKE 24581,64:POKE 24600,96).

- *Spostamento a sinistra*. Cambiare le locazioni $6021 in $40 e $6043 in $60 (usando l'istruzione POKE 24609 : POKE 24643,96).

- *Spostamento a destra*. Cambiare le locazioni $606F in $40 e $6091 in $60 (usando l'istruzione POKE 24687,64 : POKE 24721,96).

- *Spostamento in su*. Cambiare le locazioni $60BF in $44, $60C7 in $40 e $60E2 in $60 (usando l'istruzione POKE 24767,68 : POKE 24775,64 : POKE 24802,96).

- *Spostamento in giù*. Cambiare le locazioni $6117 in $5B, $611F in $5F e $613A in $40 (usando l'istruzione POKE 24855,91 : POKE 24863,95 : POKE 24890,64).

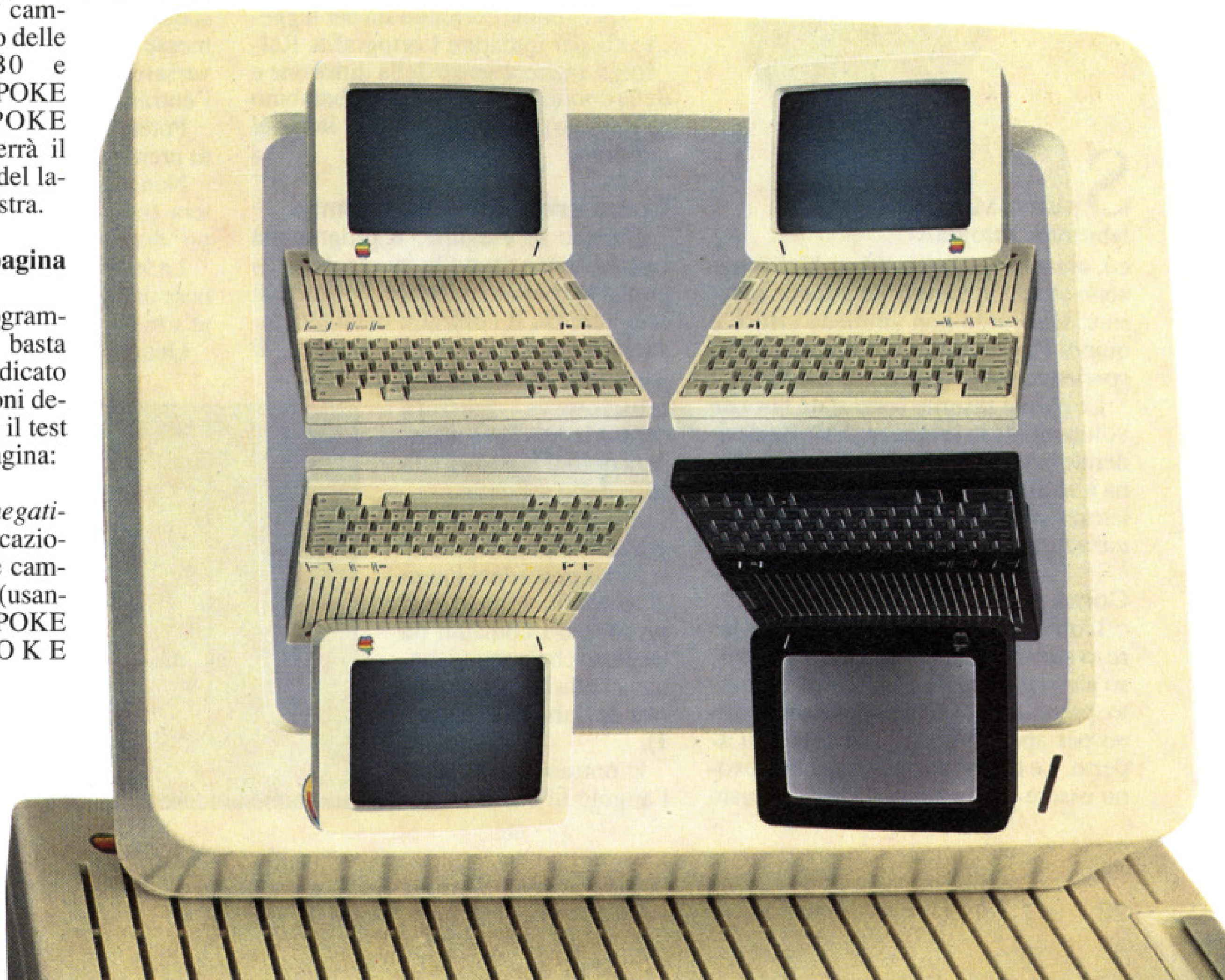
- *Inversione sinistra-destra*. Cambiare le locazioni $616D in $40 e $61CA in $60 (usando l'istruzione POKE 24941,64:POKE 25034,96).

- *Inversione alto-basso*. Cambiare le locazioni $61FD in $40, $6205 in $5F, $620D in $40, $6215 in $5F, $6235 in $60 e $627A in $60. (POKE 25085,64 :POKE 25093,95 :POKE 25101,64 :POKE 25109,95 :POKE 25141,96 :POKE 25210,96).

Per fare questi cambiamenti in Basic basta fare i POKE come indicato. Se intendete tornare alla funzione originale nello stesso programma fate attenzione a salvare i valori originali leggendoli prima con PEEK.

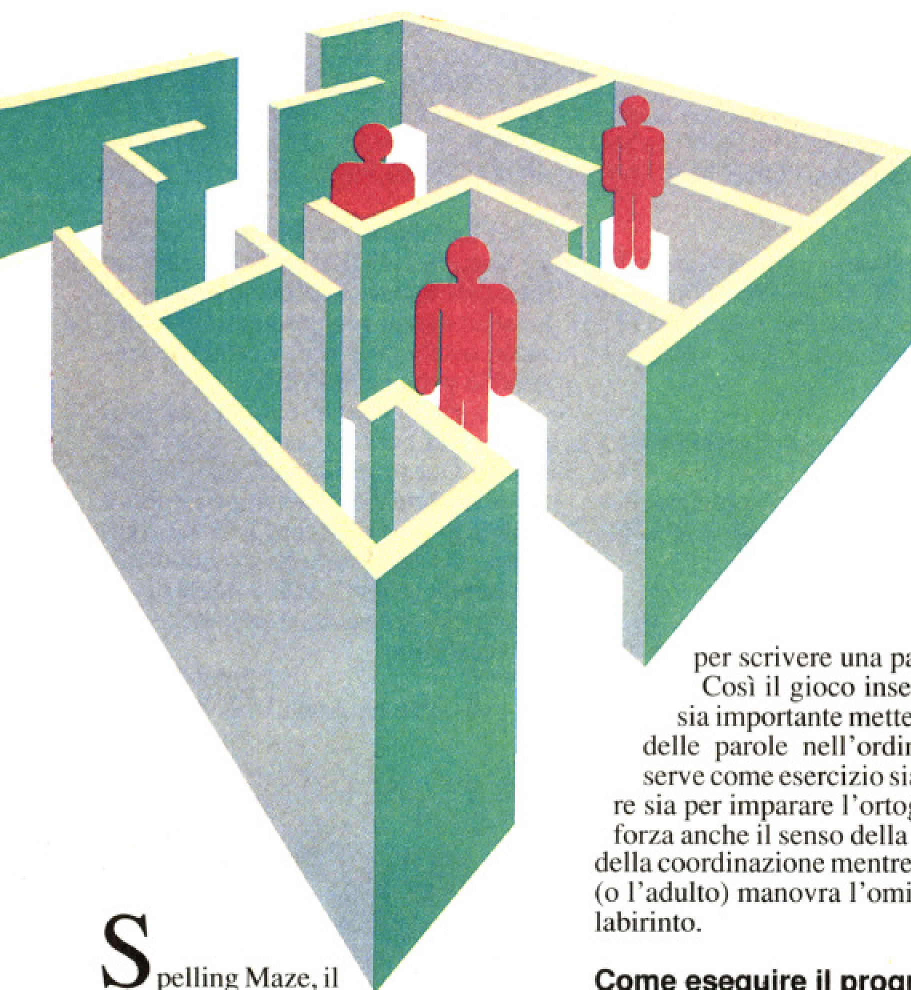
Con giudiziose modifiche del programma si possono creare molti altri effetti. Questi comprendono l'operazione su parte di uno schermo Hi-Res e l'impiego di una pagina 3 Hi-Res fantasma. Altre modifiche possono migliorare la velocità delle routine di Hi-Res Tricks a spese della memoria e della rilocabilità.

John Majka



Con questo programma il vostro bambino può divertirsi e al tempo stesso imparare il significato delle parole in inglese.

Impara giocando



Spelling Maze, il labirinto ortografico, aiuta a insegnare ai bambini come sono scritte le parole, ed è facile da usare. Il metodo non ammette errori, e quindi l'apprendimento diventa un'esperienza positiva.

Le parole inserite sono state lasciate volutamente in lingua inglese. Se desiderate inserire le parole in lingua italiana è sufficiente sostituirle seguendo le istruzioni indicate nel paragrafo Personalizzazione.

Come si usa il programma

L'obiettivo del gioco consiste nel fare avanzare l'omino raffigurato in basso a sinistra nel disegno, in un labirinto, prendendo le "chiavi" che occorrono per aprire l'uscita e sbucare all'aperto. Le chiavi sono lettere che devono essere prelevate nell'ordine giusto per scrivere una parola.

Così il gioco insegna quanto sia importante mettere le lettere delle parole nell'ordine esatto e serve come esercizio sia per leggere sia per imparare l'ortografia. Rafforza anche il senso della direzione e della coordinazione mentre il bambino (o l'adulto) manovra l'omino lungo il labirinto.

Come eseguire il programma

Quando lo eseguite, il programma caricha innanzitutto la tavola delle figure e la routine musicale. Dovete poi scegliere fra il controllo mediante tastiera e quello mediante joystick. Il programma disegnerà un labirinto,vi metterà dentro le lettere e stamperà quindi la parola alla base dello schermo.

A questo punto il programma è pronto perché si possa procedere lungo il labirinto, usando o il joystick o i comandi da tastiera, che sono visualizzati alla base e alla destra del labirinto (**figura 1**).

L'omino parte dall'angolo in basso a sinistra del labirinto. Premete i tasti I, J, K o M (oppure usate il joystick) per spostarlo in su, a sinistra, a destra o in giù (i comandi sono ammessi sia in maiuscolo sia in minuscolo). Non sono permesse mosse non valide, come attraversare un muro o cercare di uscire dall'entrata.

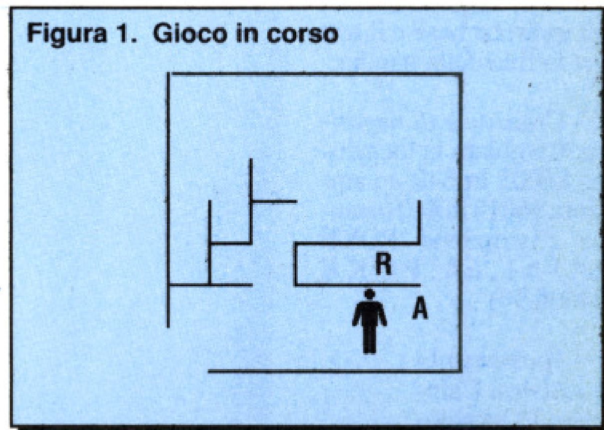
Potete smettere in qualsiasi momento premendo il tasto Escape.

Non appena avrete raggiunto una lettera (nell'ordine esatto) si sentirà un po' di musica e la lettera scomparirà.

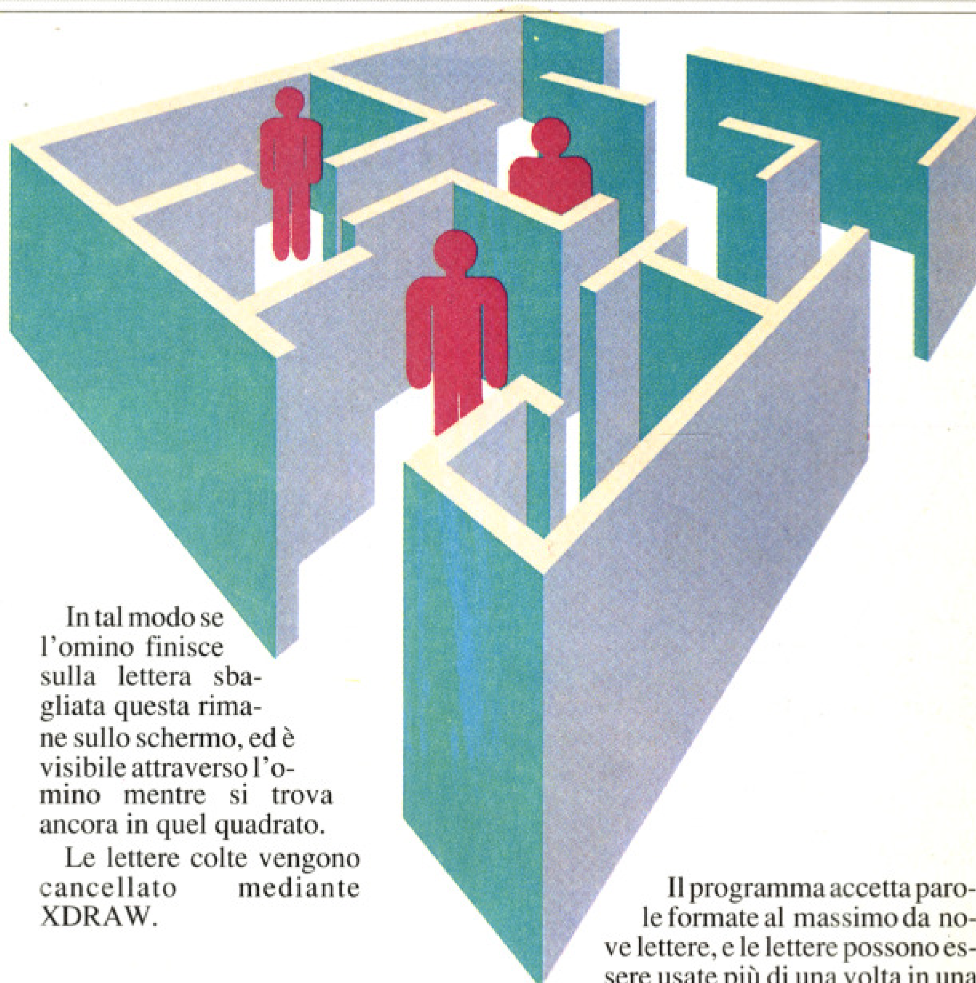
Le lettere della parola stampata alla base dello schermo saranno evidenziate a mano a mano che verranno colte.

Quando sarà stata colta l'ultima let-

Figura 1. Gioco in corso







tera si sentirà altra musica e l'uscita nell'angolo in alto a destra del labirinto si spalancherà.

A questo punto potrete spostare l'omino attraverso l'uscita per mettere fine al gioco.

Quando avrete completato con successo il percorso nel labirinto si sentirà altra musica e il programma vi chiederà se volete giocare ancora.

Premendo qualsiasi tasto diverso da Q si ricomincerà il gioco con un nuovo labirinto e una nuova parola. Se usate il joystick, uno qualunque dei due pulsanti farà ricominciare il gioco.

La stessa parola verrà ripetuta (sempre in ordine casuale)solo quando tutti i vocaboli compresi nella lista saranno stati utilizzati. Premendo Q si mette fine alla seduta.

Funzionamento del programma

La più complessa delle 14 routine di Spelling Maze (**tavola 1**) è la routine che genera il labirinto nelle linee 550-780. Ogni quadrato del labirinto ha un elemento di matrice che mostra lo stato di ciascun muro (matrici UP, DN, LT, RT di questo programma).

Un'altra parte importante del programma è l'impiego di XDRAW per mettere l'omino sullo schermo in modo che le lettere sulle quali va a posarsi non siano cancellate.

In tal modo se l'omino finisce sulla lettera sbagliata questa rimane sullo schermo, ed è visibile attraverso l'omino mentre si trova ancora in quel quadrato.

Le lettere colte vengono cancellato mediante XDRAW.

Il programma accetta parole formate al massimo da nove lettere, e le lettere possono essere usate più di una volta in una parola.

Potete anche cambiare i tasti che spostano l'omino.

La routine delle linee 500-540 predispone i comandi e quindi tutto quello che dovete fare è di mettere la maiuscola che preferite per i comandi in su, in giù, a sinistra e a destra fra le virgolette, rispettivamente, delle linee 510, 520, 530 e 540.

I simboli appropriati (lettere o frecce) saranno disegnati a destra del labirinto, per rammentarvi i comandi durante il gioco.

Potreste anche prendere in considerazione l'aggiunta al labirinto di lettere estranee alla parola che, distraendo il giocatore, renderebbero più difficile il completamento della parola esatta. Potreste anche togliere la parola che è alla base dello schermo, ma in tal caso, per trovare la grafia esatta, il giocatore dovrà anche ricostruire la parola lettera per lettera.

Soprattutto, cercate di commisurare il livello di abilità del gioco alle capacità del vostro bambino.

Tavola 1. Le routine di Spelling Maze

Linee	Routine
10-370	Allestimento iniziale
380-490	Riceve la parola per questa partita
500-540	Predispone i comandi
550-780	Disegna il labirinto
790-890	Mette le lettere nel labirinto
900-1070	Mette l'uomo nel labirinto e le istruzioni sullo schermo
1080-1250	Riceve il comando e lo esegue
1260-1330	Controlla se è stata colta la lettera esatta
1340-1380	Apre l'uscita se appropriato
1390-1450	Controlla se si è "vinto"
1460-1520	Routine di abbandono del gioco
1540-1770	Routine musicali
1780-1880	Routine d'errore
1890-2010	Routine di controllo da joystick

Personalizzazione

Di tutti i modi di personalizzazione del programma il più ovvio è l'aggiunta (o la sostituzione) di queste parole per il vostro bambino. Il numero di parole (variabile NW) è fissato a linea 120, e quindi basta cambiare questo numero e aggiungere le nuove parole all'istruzione DATA di linea 310.

William P. Doyle, Jr.

Una delle cose peggiori che possa capitare a un programmatore è quella di cancellare da un dischetto, per sbaglio, un programma buono, magari appena fatto e del quale non ha un'altra copia. Oppure chiamare il Catalog e vedersi scorrere sul monitor un Catalog infinito che ripete sempre gli stessi sette file.
Per rimediare a tali situazioni serve un programma in grado di leggere, scrivere, ma soprattutto gestire i settori del dischetto in questione e che in più....

Disco detective



Il programma che ci vuole si chiama Explorer. Gira su Apple II sotto DOS, è scritto quasi tutto in Basic e consiste in un unico listato lungo circa 10 Kb; inoltre, per facilitarne l'uso, ad ogni schermata vengono visualizzati i tasti da usare per eseguire le possibili operazioni. Sostanzialmente il programma consiste in una serie di procedure per l'esplorazione dei dischetti, che vanno a formare tre operazioni principali:

- esplora Catalog
- esplora Settori
- esplora File.

Queste sono in grado non solo di leggere, ma anche di modificare e riscrivere i settori desiderati.

Vediamole separatamente una alla volta:

• Esplora Catalog.

Questa subroutine permette di leggere e modificare i dati della directory del dischetto, evidenziando i file cancellati ma recuperabili. La lista dei settori del Catalog parte normalmente dal settore15 della traccia 17 ed è così organizzata:

traccia 17, settore 15 :
byte
1: prossima traccia (esempio 17)
2: prossimo settore (esempio 14)
11...45: dati file 1
46...81: dati file 2
:
...... dati file 7

traccia 17, settore 14:
byte
1: prossima traccia (esempio 0) - fine
2: prossimo settore (esempio 0) - fine
11...45: dati file 1
46...81: dati file 2
:
... dati file 7

Quando i puntatori sono a zero significa che la lista è finita. Ogni settore contiene i dati relativi a 7 file e ad ogni file sono dedicati 35 bytes ripartiti così:

byte (dei 35)
0: traccia di partenza del file
1: settore di partenza del file
2: tipo di file
3...32: filename
33: numero di settori occupati
34: vecchio numero di traccia se il file è cancellato.

Il byte 34 ci permette di recuperare un file cancellato scrivendo il suo valore nel byte 0. Il settore 0 della traccia 17 contiene le informazioni riguardanti il disco.

Tutto questo e altro ancora è spiegato da pag. 367 a pag.370 del manuale Apple II Guida all'uso (appendice H).

La subroutine gestisce tutti questi dati, permettendo la loro modifica. Se, come nell'esempio dell'introduzione, un dischetto presenta un Catalog che ripete sempre gli stessi 7 file, di sicuro ci sarà un settore della directory che punta su se stesso. Per esempio:

traccia 17, settore 15:
byte
1: prossima traccia = 17
2: prossimo settore = 15

significa che dopo il settore 15 viene ancora il settore 15 e poi ancora lo stesso, e così via. Basterà allora modificare il puntatore al prossimo settore con il numero corretto del settore successivo e siccome di solito il Catalog va dal settore 15 al settore 14 (della traccia 17) in questo caso basterà scrivere 14 anziché 15 sul puntatore.
Per resuscitare un file cancellato, basterà scrivere sul puntatore alla sua traccia di partenza il valore del byte 34 dell'area del settore dedicatagli. Quando un file viene cancellato, infatti, viene posta a 255 il puntatore alla sua traccia iniziale, che viene però salvato prima nel byte 34, per poterlo eventualmente recuperare.Ma tutto questo lo fa automaticamente il programma; i tasti da usare, come già detto, sono di volta in volta spiegati ad ogni schermata.

• Esplora Settori.

Il punto di forza del programma consiste nella possibilità di leggere un settore, modificarlo e riscriverlo. Questo viene fatto con l'opzione Esplora settori, richiamabile dal menù con il tasto S e usata anche dalla successiva opzione Esplora file. Consiste in due fasi: Posizionamento e Edit.

Con la prima fase ci si posiziona nel settore e nella traccia desiderata, specificandoli o scegliendoli con le altre due opzioni (poiché Esplora settori si posiziona inizialmente sull'ultimo settore che il programma ha letto). Da questa col tasto E si entra nella fase di Edit della prima metà del settore e vengono visualizzati i relativi comandi; il settore è stato diviso in due parti, perché non rientrava tutto in una sola schermata.

Fatte le modifiche e tornati (con Esc) alla fase di Posizionamento, il settore si può salvare o rileggere, nel caso le modifiche fatte fossero sbagliate. Sempre con Esc, si torna ancora al menù.

• Esplora File.

Esplora settori è usato anche da Esplora file. In sostanza, questo comando non è altro che un'esplorazione dei settori, pilotata secondo il percorso traccia-settore di un determinato file.

Per prima mossa ci si posiziona sul file da esplorare, poi il programma legge la lista (una sorta di percorso) delle tracce e dei settori contenenti i dati del file (vedi pag. 367 del manuale).Quindi, questi settori possono essere visti ed modificati uno dopo l'altro.

• Mappa settori.

Oltre alle tre procedure appena viste, ne esiste una quarta che visualizza la disponibilità dei settori di un disco.

La mappa dei settori occupati è contenuta nel settore 0 nella traccia 17, a partire dal byte 59 (4 byte per traccia, di cui 2 non usati). Un bit a 0, significa che il settore è occupato, a 1 significa che è libero.

OBJ Explorer

Il programma si serve di due brevi routine in linguaggio macchina, contenute sotto forma di data, nel programma Basic.

La prima permette di richiamare la routine RWTS (Read Write Track Sector) e contiene la File Manager Parameter List (lista dei parametri).La seconda stampa sul monitor a velocità LM il contenuto del buffer, dove vengono caricati i settori ed è usata sia da Esplora settori sia da Esplora file. Si è resa necessaria per poter scorrere velocemente i settori di un file o di una traccia. Entrambe le routine si trovano nella zona libera della pagina 2 della memoria (a partire da $300 esadecimale).

Menù principale

Il menù principale comprende in totale sette opzioni.Quattro di queste le abbiamo già elencate e si richiamano con i tasti C, S, F e M; per quanto riguarda le altre
• Esc fa terminare il programma
• D serve per cambiare drive
• R serve per ripartire, nel caso si cambiasse il disco ma non il drive.

Marco Busetto

Convertite qualsiasi numero in notazione scientifica.
Questa breve routine in Applesoft
vi permette di specificare il numero delle cifre significative.

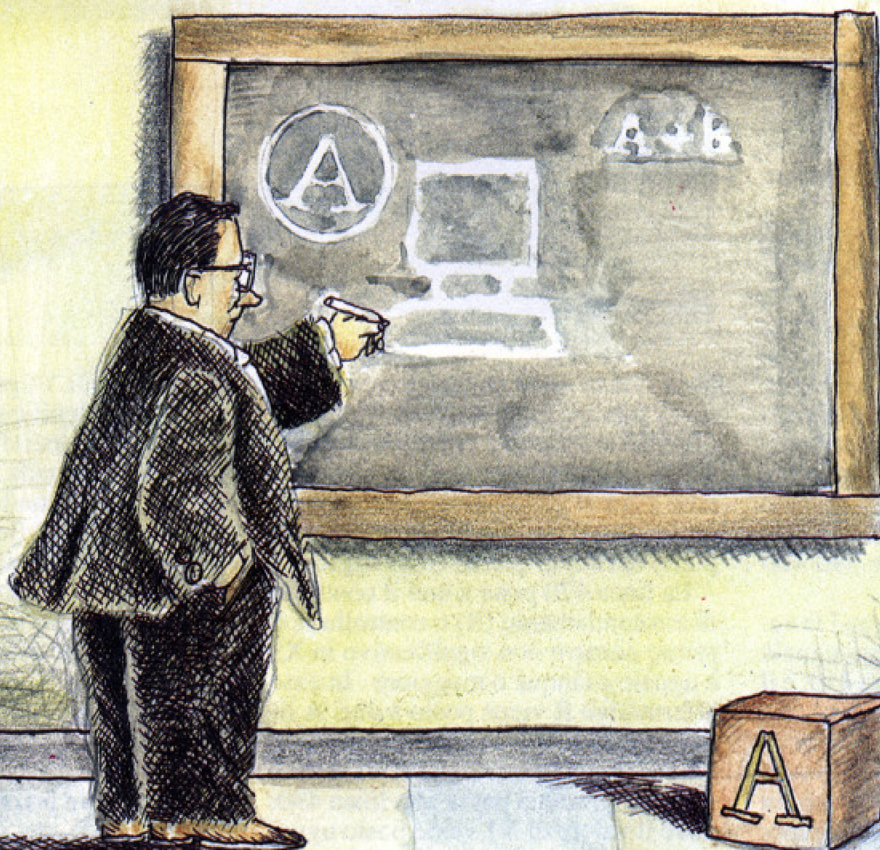
Formattazione per notazione scientifica

Le applicazioni scientifiche utilizzano numeri che spaziano da molto piccoli a molto grandi.

Quando i numeri sono il risultato di misure sperimentali il numero deve riflettere la precisione della misurazione. La notazione esponenziale (o notazione scientifica, come spesso viene chiamata) risolve elegantemente questo problema esprimendo i numeri mediante un sistema che si basa su potenze di dieci.

È comunemente usata, per esempio, nei calcoli chimici e fisici.

L'impiego della stampa per la notazione scientifica compie rapide conversioni in notazione scientifica. Presentiamo in questo numero un programma esemplificativo comprendente la routine per l'utilizzo di questa funzione.



9.99999999 1.34x10-2 8.7x101

Come si usa il programma

La notazione scientifica esprime un numero come coefficiente e una potenza di dieci.

Così nella notazione scientifica il numero 87 sarebbe scritto nella forma 8.7x101, -0.0134 sarebbe -1.34x10-2 e 9 sarebbe 9x100.

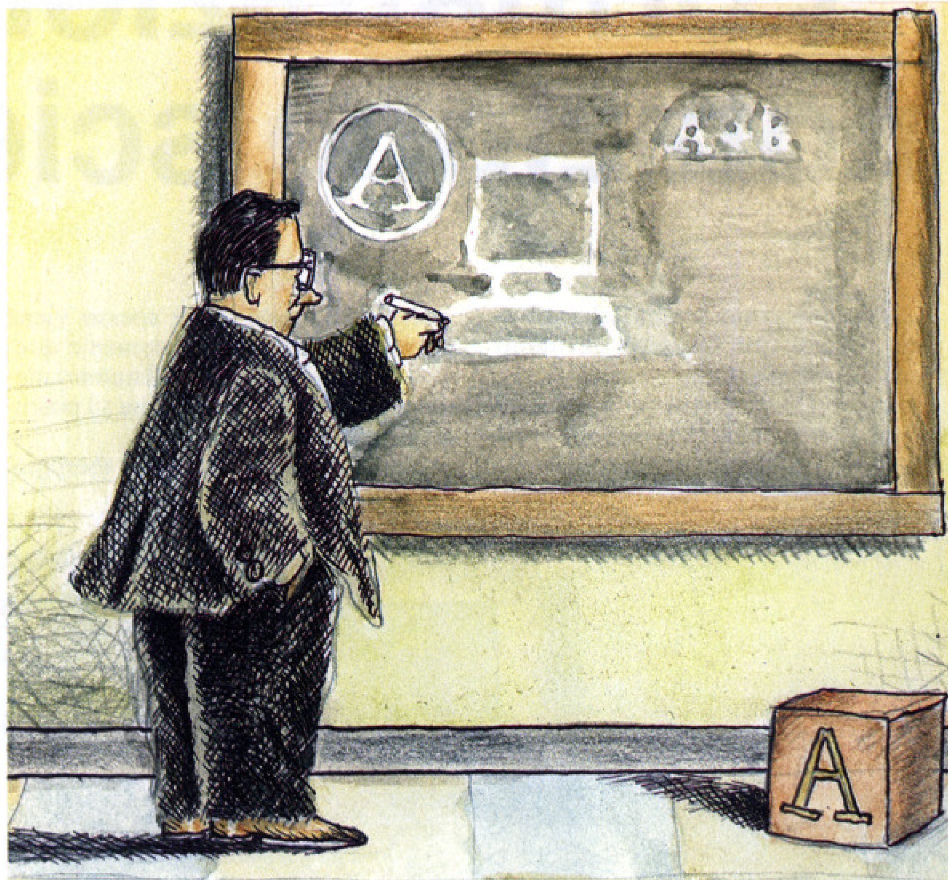
Le cifre significative costituiscono un metodo per esprimere la precisione di un numero. Per esempio 1.72 a due cifre significative è 1.7, mentre 81.76 a tre cifre significative è 81.8. La regola è che se la prima cifra non significativa di un numero è uguale o maggiore,a cinque il numero viene arrotondato.

Funzionamento del programma

Le linee 10-250 contengono il programma dimostrativo. La linea 80 istituisce un procedimento di gestione degli errori, mentre le linee 90-110 predispongono il display.

La linea 120 vi chiede di introdurre un numero (X) e la linea 130 chiede il numero delle cifre significative (P). La linea 140 controlla che il numero delle cifre significative sia compreso fra uno e nove. Se questa condizione è soddisfatta il programma salta alla subroutine d'impiego della stampa. Al ritorno dalla subroutine la linea 200 stampa la risposta in questa forma: coefficiente per 10x.

Le linee 210-250 vi permettono di introdurre un altro numero o di mettere fine al programma.



La subroutine d'impiego della stampa è situata nelle linee 260-530. Le linee 260-340 sono istruzioni REM che listano le variabili richieste dalla subroutine (X e P), ritornate dalla subroutine (Y$ e NC) e usate nella subroutine (X1, X e R). Le linee 350-360 trattano il caso di X = 0, le linee 370-380 trattano il caso di X = 1 e le linee 390-400 trattano il caso di X = -1.

Se X non è uguale a 1, a 0 o a -1 la variabile fittizia, X1, viene posta uguale al valore assoluto di X, a linea 410, e il contatore d'esponente, NC, viene inizializzato a zero. Il cuore della subroutine è nelle linee 420 e 430. X1 viene convertito in un numero compreso fra 1 e 9.99999999, e il contatore d'esponente viene aumentato o diminuito di uno a ogni passaggio.

Le linee 440 e 460 creano le stringhe X$ (l'equivalente stringa di X1) e Y$ (che è X$ al giusto numero di cifre significative). Se la stringa è priva di punto decimale questo viene aggiunto.

La linea 470 pone a uno il controllo di arrotondamento (R) e controlla se il primo numero non significativo in X$ è uguale a cinque o maggiore . In caso affermativo R viene posto a due. A linea 490, se R = 1 il flusso di programma passa alla linea 520, se R = 2 il flusso di programma passa alla linea 490, dove il valore di X1 viene posto uguale al valore di Y$ più uno nella sua ultima cifra significativa. La linea 500 arrotonda i numeri fino a dieci (così per esempio arrotonda 9.99 a due cifre significative portandolo a 10).

A linea 510 il flusso di programma torna indietro saltando a linea 440. A linea 520 se X è minore di zero viene aggiunto un segno di meno all'inizio di Y$. La linea 530 riporta al programma principale.

Questo programma può essere utilmente inserito come subroutine nei vostri programmi di calcolo per ottenere una maggiore precisione.

George W. Goth

Continua il corso sull'uso del sistema operativo DOS 3.3 e del Basic Applesoft.
In questa puntata viene spiegata l'organizzazione del dischetto.

DOS 3.3 senza veli

Nella prima puntata abbiamo visto quali sono i principali comandi del Dos 3.3. In questa seconda puntata vedremo come vengono registrati i files su disco.

Occorre innanzitutto conoscere come è fatto un dischetto: all'atto della formattazione il computer divide il disco in 35 tracce concentriche, dovute allo spostamento assiale della testina; a sua volta ogni traccia è divisa in 16 settori. Questa suddivisione serve al computer come riferimento, per sapere esattamente dove sta leggendo o scrivendo.

Possiamo paragonare un disco formattato a un foglio a righe: così come le righe servono per scrivere dritto, le tracce e i settori servono al computer per registrare i dati in modo ordinato.

A questo punto sorgono alcuni problemi:

1) come fare a distinguere un settore da un altro? Cioè, se si deve leggere il terzo settore della quinta traccia, come distinguerlo dagli altri 15? Si dovrà far precedere il settore da una serie di dati che indicano il numero del settore stesso e quello della traccia cui appartiene. Questa serie di dati è comunemente chiamata campo indirizzi (address field).

Non solo, per evitare la sovrapposizione di un settore con un altro, a causa della velocità del drive che può variare leggermente o dell'utilizzo dello stesso disco su drives differenti, è stato lasciato uno spazio di sicurezza (GAP) fra un settore e l'altro e fra il campo indirizzi ed il campo dati. Il settore è praticamente formato in questo modo:

/gap/campo indirizzi/gap/campo dati/gap/

Ripetete per 16 volte e avrete una traccia.

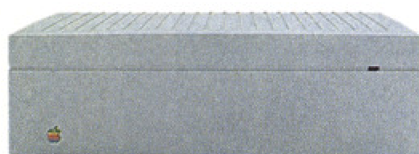
2) Tutti i dati sono registrati effettivamente su disco in forma binaria (10110010....) ma, poiché l'hardware ha diverse idiosincrasie, un byte su disco non può assumere tutti i 256 valori di un byte in memoria: quindi è richiesta una particolare codifica dei singoli dati che, però, richiede più di un byte-disco per ogni byte-memoria. Questo tipo di codifica è chiamata "nibblizzazione" (assurdo onomatopeico derivato dal nome inglese del singolo byte-disco codificato: nibble).

I sistemi di codifica sono due: uno per il campo indirizzi, chiamato codifica quattro e quattro, e uno per il campo dati chiamato codifica sei e due. Niente paura: non ci addentreremo troppo tra byte e bit. È sufficiente sapere che il sistema operativo automaticamente "nibblizza" i dati quando scrive un settore e li "denibblizza" quando lo legge riportandoli al loro stato naturale.

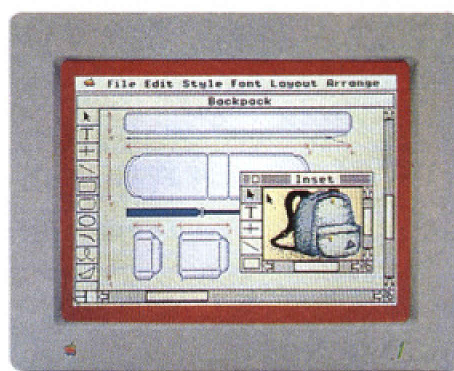
3) Visto che tutti i valori sul disco sono in binario, c'è solo una probabilità su otto che il primo bit letto dalla testina del drive sia l'inizio di un byte.

Per risolvere questo problema sono stati usati, nei gap, dei bytes particolari, chiamati Self-sync. Questi bytes hanno la caratteristica di avere due bit in più, entrambi a zero, e di valere generalmente 256 (in esadecimale $FF).

Proprio per le caratteristiche del sistema operativo (il primo bit di un nibble deve essere 1, se no salta al successivo bit), la sincronizzazione viene raggiunta dopo aver letto un massimo di 5 self-sync (**tavola 1**).







Bisogna aggiungere che sia il campo indirizzi sia il campo dati sono caratterizzati da tre bytes iniziali (prologo) e da tre bytes finali (epilogo) che li distinguono dagli altri dati.

Nibble per nibble il settore è costituito come visibile nella **tavola 2.**

Ogni settore è composto da una serie di self-sync, cui seguono tre nibble di identificazione del campo indirizzi (prologo), i numeri del volume, della traccia, del settore, il checksum di questi dati e tre nibbles di fine campo indirizzi (epilogo). Seguono sei self-sync, tre nibbles di identificazione campo dati (prologo), i dati effettivi, il checksum e tre nibbles di fine campo dati (epilogo).Il checksum è un numero ricavato dalla somma dei bytes precedenti; ogni volta che il sistema legge il campo riesegue la somma e la controlla con quella letta dal disco. Se sono uguali continua la lettura, altrimenti si ferma perché considera i dati non affidabili e restituisce l'odiato (e temuto) I/O ERROR.

A questo punto, in teoria, dovremmo avere un disco con 32 * 16 = 560 settori disponibili, ma in effetti bisogna togliere lo spazio occupato dal DOS (tre tracce, dalla zero alla due) e la traccia 17 dove risiedono il catalogo e la mappa di occupazione del disco (o VTOC).

Vediamo intanto cosa succede all'interno del computer all'atto dell'accensione. Il sistema effettua un autocontrollo, quindi cerca negli slot, partendo dal 7 e andando indietro (questo è il motivo per cui chi ha i dischi da 3",

Tavola 1.

```
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/ (----------)  (----------)  (----------) (----------)  (----------)  (----------) (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/  (----------) (----------)  (----------) (----------)  (----------)  (----------) (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/    (----------)(----------)   (----------) (----------)  (----------)  (----------) (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/     (----------)(----------) (----------) (----------)  (----------)  (----------)  (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/       (----------)(----------)(----------) (----------)  (----------)  (----------)  (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/        (----------)(----------)(----------) (----------)  (----------) (----------)  (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/         (----------)(----------)(----------) (----------)  (----------) (----------)  (----------)   /
/ 1111111100111111110011111111001111111100111111110011111111001111111100 /
/          (----------)(----------)(----------)(----------) (----------)  (----------)  (----------)  /
```

Tavola 2.

```
-------------------------------------------------------------------------------------------
/  FF FF FF   / D5 AA 96 /  XX XX  /   YY YY  /  ZZ ZZ   /  CC CC  / DE AA EB /
-------------------------------------------------------------------------------------------
/ SELF-SYNC / PROLOGO /N. VOLUME/N. TRACCIA/N. SETTORE/CHECKSUM/ EPILOGO /
-------------------------------------------------------------------------------------------
/    GAP    /                    CAMPO  INDIRIZZI                              /
-------------------------------------------------------------------------------------------


-------------------------------------------------------------------------------------
/  FF FF FF   / D5 AA AB /                          /    CC    / DE AA EB /
/------------------------------------------------------------------------------------
/ SELF-SYNC / PROLOGO /  DATI  ( 342 NIBBLES)  /CHECKSUM/ EPILOGO /
-------------------------------------------------------------------------------------
/    GAP    /                   CAMPO   DATI                          /
-------------------------------------------------------------------------------------
```

generalmente nello slot 5, all'accensione vede partire prima i dischi da 5" 1/4,e poi gli altri) se c'è un drive.

Se lo trova, come di solito, nello slot 6, lancia la routine che si trova alla locazione $C600 (si usa quasi più l'esadecimale del decimale, tanto che le routine delle singole schede risiedono in $Cx00, dove x è il numero di slot in cui è inserita la scheda). Provate ad andare in monitor con

CALL-151

ed inserite

C600G....

La routine in $C600 carica il contenuto del settore zero della traccia zero (da questo momento tracce e settori verranno indicati rispettivamente con T e S, per esempio T0-S0), chiamato bootstrap, in $800, eseguendolo. Questo carica la routine principale del DOS, quella che materialmente legge e scrive sul disco (RWTS, Read Write Track Sector, Scrittura e lettura di traccia e settore), posizionata nei settori dall'uno al nove di traccia zero, in locazione $B700. La RWTS carica finalmente il resto del DOS, in pratica l'interprete dei comandi, che carica e lancia il programma che era in memoria all'atto della formattazione del disco (generalmente Hello).

Il Dos 3.3 è diviso, sul dischetto, in tre parti: il bootstrap (T0-S0), la routine RWTS, e il DOS. Anche in memoria la routine RWTS rimane a sé e può essere chiamata direttamente, senza usare l'interprete del DOS, nella prossima puntata vedremo come fare.

Francesco Adornato

(continua)

